

PLATO: A COORDINATION FRAMEWORK FOR DESIGNERS OF MULTI-PLAYER REAL-TIME GAMES

A Thesis Submitted to the College of
Graduate Studies and Research
in Partial Fulfillment of the Requirements
for the Degree of Masters of Computer Science
in the Department of Computer Science
University of Saskatchewan
Saskatoon

By
Ariyan Zohoorian

© Ariyan Zohoorian, April 2013. All rights reserved.

PERMISSION TO USE

In presenting this thesis in partial fulfilment of the requirements for a Postgraduate degree from the University of Saskatchewan, I agree that the Libraries of this University may make it freely available for inspection. I further agree that permission for copying of this thesis in any manner, in whole or in part, for scholarly purposes may be granted by the professor or professors who supervised my thesis work or, in their absence, by the Head of the Department or the Dean of the College in which my thesis work was done. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to the University of Saskatchewan in any scholarly use which may be made of any material in my thesis.

Requests for permission to copy or to make other use of material in this thesis in whole or part should be addressed to:

Head of the Department of Computer Science

University of Saskatchewan

Saskatoon, Saskatchewan

Canada

S7N 5C9

ABSTRACT

Player coordination is a key element in many multi-player real-time digital games and cooperative real-time multi-player modes are now common in many digital-game genres. Coordination is an important part of the design of these games for several reasons: coordination can change the game balance and the level of difficulty as different types and degrees of coordination can make the game easier or more difficult; coordination is an important part of ‘playing like a team’ which affects the quality of play; and coordination as a shared activity is a key to sociality that can add to the sociability of the game. Being able to exercise control over the design of these coordination requirements is an important part of developing successful games. However, it is currently difficult to understand, describe, analyze or design coordination requirements in game situations, because current frameworks and theories do not mesh with the realities of video game design. I developed a new framework (called PLATO) that can help game designers to understand, describe, design and manipulate coordination episodes. The framework deals with five atomic aspects of coordinated activity: Players, Locations, Actions, Time, and Objects. PLATO provides a vocabulary, methodology and diagram notation for describing and analyzing coordination. I demonstrate the framework’s utility by describing coordination situations from existing games, and by showing how PLATO can be used to understand and redesign coordination requirements.

ACKNOWLEDGMENTS

I would like to thank my supervisors Dr. Carl Gutwin and Dr. Kevin Stanley whose support, knowledge and patience made this thesis possible.

CONTENTS

1	INTRODUCTION	1
1.1	PROBLEM	2
1.2	MOTIVATION.....	3
1.3	SOLUTION	4
1.4	EVALUATION	4
1.5	CONTRIBUTION	5
1.6	OUTLINE OF THESIS	5
2	LITERATURE REVIEW	6
2.1	COORDINATION.....	6
2.2	GAME THEORY	8
2.3	COORDINATION AND AWARENESS	10
2.4	COORDINATION IN DIGITAL MULTIPLAYER GAMES	11
2.5	COORDINATION IN KINESIOLOGY	13
2.6	COOPERATIVE GAME DESIGN PATTERNS	13
2.7	COLLABORATIVE LEARNING	15
2.8	SOCIAL INTERACTION DESIGN	17
2.9	MOBILE MIXED REALITY (MMR) GAMES	22
3	PLATO: A FRAMEWORK OF COORDINATION IN GAMES.....	23
3.1	PLATO PART 1: PLAYERS	24
3.1.1	<i>What game entities can act as a player?</i>	24
3.1.2	<i>How are players specified?</i>	25
3.2	PLATO PART 2: LOCATIONS	26
3.3	PLATO PART 3: ACTIONS	26
3.4	PLATO PART 4: TIME	27
3.4.1	<i>Temporal classifications</i>	28
3.5	PLATO PART 5: OBJECTS.....	30
3.5.1	<i>Object types</i>	30

3.6	GOALS AND INTERDEPENDENCIES	32
4	ANALYZING COORDINATION	34
4.1	PLATO AS AN ANALYTIC TOOL	35
4.2	EXAMPLE: COMMANDOS.....	37
4.3	GAME BALANCE ADJUSTMENTS	40
5	COORDINATION IN PORTAL2 CO-OP MODE	43
5.1	EXPERIMENTAL SETUP	43
5.2	INTERACTION TOOLS IN PORTAL 2	44
5.3	GAME ELEMENTS	44
5.4	RESULTS	45
5.4.1	<i>Portal2 - Scenario 1 (1, 1, 1)</i>	46
5.4.2	<i>Portal2 - Scenario 2 (1,2,2)</i>	49
5.4.3	<i>Portal2 - Scenario 3 (1,3,2)</i>	51
6	PLATO AS A COORDINATION DESIGN TOOL.....	54
6.1	STEALTH HACKER.....	56
6.1.1	<i>Stealth Hacker's gameplay evaluation</i>	57
6.2	STEALTH HACKER V2.0 (SH2)	59
6.3	STEALTH HACKER ANALYSIS.....	60
6.3.1	<i>Coordination design by general gameplay manipulation</i>	61
6.3.2	<i>Coordination design at certain milestones</i>	65
7	DISCUSSION AND DIRECTIONS FOR FUTURE STUDY	73
7.1	THE UNIQUE APPROACH OF THE PLATO DIAGRAM NOTATION.....	75
7.2	SPECIFIC FEATURES OF PLATO DIAGRAM NOTATION TOOLS	78
8	CONCLUSION.....	82
8.1	SUMMARY	82
8.2	CONTRIBUTIONS	83
8.3	FUTURE WORK	84
9	REFERENCES	86

LIST OF TABLES

Table 2-1. Prisoners' dilemma payoff matrix.....	9
Table 2-2. Stag hunt payoff matrix.....	9
Table 3-1. Player specification summary.....	26
Table 3-2. Different types of objects and their specifications	32
Table 3-3. PLATO Framework summary.....	33
Table 3-4. Elements of the framework	33
Table 3-5. Interdependencies in PLATO	33
Table 4-1. Time elements and their diagram notations.....	35
Table 5-1. Coordination elements for scenario (1, 1, 1).....	46
Table 5-3. PLATO elements for scenario (1, 3, 2)	52

LIST OF FIGURES

Figure 2-1. “higraphs” as a methodology of diagramming in the form of finite-state machine...	18
Figure 4-1. An schema of the hallway scenario.....	36
Figure 4-2. PAT (top) and LPT diagrams for the dark hallway episode. L1 represents the location where the first switch is located and L2 refers to the location of the second switch.....	37
Figure 4-3. Coordination episode in Commandos, showing locations of soldiers (L3), their views (triangles), cigarette (L2) and commandos (L1) at the beginning of the episode.....	38
Figure 4-4. PAT diagram for Commandos sub-scenario	39
Figure 4-5. LPT diagram for Commandos episode. Locations are illustrated with different dashed lines. Circles in the diagram are used as bridges over the other lines.	40
Figure 4-6. Timing error collaboration design. The thick dotted line represents the return of the first soldier which crosses over the dashed line indicating Green Beret movements.....	41
Figure 4-7. The Commandos episode when Marine has the ability to both harpoon and carry dead bodies.....	42
Figure 5-1. Game elements of Portal 2	45
Figure 5-2. Game environment of the scenario (1, 1, 1). Cameras (C1 and C2) demonstrate two views of the environment from different positions.....	46
Figure 5-3. Scenario (1, 1, 1) - PAT diagram	48
Figure 5-4. Scenario (1, 1, 1) – LPT diagram.....	48
Figure 5-5. Game environment of the scenario (1, 2, 2). Cameras (C1 and C2) demonstrate two views of the environment from different positions.....	49
Figure 5-6. Scenario (1, 2, 2) - PAT diagram	50
Figure 5-7. Game environment of the scenario (1, 3, 2). Cameras (C1 and C2) shows two renders of the environment from different positions..	51
Figure 5-8. Scenario (1, 3, 2) - PAT diagram	53
Figure 6-1. In this figure, on the right, cop's interface on a smartphone and on the left, hacker's interface on pc is illustrated.	57
Figure 6-2. PLT diagram in a single run of the Stealth Hacker game. Each player (p1, p2 ,p3) are specified with a distinct line pattern.	59
Figure 6-3. LPT diagram of an hypothetical desired game trend.	62

Figure 6-4. Stealth Hacker PAT diagram of an expected game current by using a shared resource	64
Figure 6-5. Tracking hacker's location	65
Figure 6-6. An hypothetical simple coordination design	67
Figure 6-7. An hypothetical more complex coordination design	68
Figure 6-8. The third attempt in redesigning the scenario	69
Figure 6-9. Coordination design for the two idle players shown in Figure 6-2.....	70
Figure 6-10. An overview of the three-player activities in this new scenario.	70
Figure 6-11. LPT diagram representing the spatial attributes of this scenario.	71
Figure 7-1. Gantt (top) and PERT network (bottom) charts of the hallway example in section 4.1. The dotted arrow in the Gantt chart shows the dependency of an action to its initiative activity.	78
Figure 7-2. An arbitrary generalization example of PLATO notation using highgraph.....	80

CHAPTER 1

INTRODUCTION

Games have always been an inherent part of human societies and play a key role in development of our culture and civilization [31]. Traditional games, played in the real world, by supporting interaction among players and the environment [39] are collective in nature [70]. Digital games provide players with more fantasy, challenge, and curiosity [39] but are often individual with limited collective actions [70] and constrain players' interaction. Multiplayer games which leverage player collaboration and interaction have become an active research area and are both socially and commercially important.

Games from a variety of genres supporting either collocated or networked play have achieved commercial success by supporting player collaboration and interaction [42]. Hundreds of thousands players join massive multiplayer games to participate in a social experience which encourages interactions [17]. This popularity stems from the shared experience supported by the game environment which leverages player interaction: "What makes a difference for many is apparently the shared experience, the collaborative nature of most activities and, most importantly, the reward of being socialized into a community of gamers and acquiring a reputation within it [17, p1]".

In a game scenario where players collaborate on performing a shared task and interact in a common context, player coordination leverages the game sociability: "key to sociality is coordination, that is, the bringing of people into a common action, movement or condition [62, p1]." This highlights the role of game designer as social architects [53]. Brown and Bell in a study of players' social interaction and design principles in a game called "There" explained the

player's social activities as: "There involves coordination, since one player needs to drop the car from high up, while the second player jumps into the car as it falls. The pleasure of this activity comes in part from the difficulty in coordinating actions together. This suggests a key design goal for future multiplayer games will be supporting in game social activities [6, p8]."

Cooperative real-time multi-player modes are now common in many video-game genres, including first-person shooters (e.g., Rainbow Six), platformers (e.g., LittleBigPlanet), puzzle games (e.g., Portal 2), and sports games (e.g., FIFA Soccer). In collaborative games, players must coordinate their actions to achieve goals and objectives. In a shooter, for example, one player might have to throw a grenade into a room at the exact moment that another player kicks open the door, while a third player draws enemy fire. Similarly, players in Portal 2's co-op mode must carefully synchronize the placement of portals and objects to escape from a maniacal AI opponent. Coordination is an important part of the design of these games, for several reasons: different types and degrees of coordination can make the game easier or more difficult; coordination is an important part of 'playing like a team' and is a skill that can provide better players an advantage; and coordination is a shared activity that can add to the sociability of the game [59].

1.1 Problem

In many of the work domains studied in previous research, management of interdependencies is required to achieve productivity, but in multiplayer games, the *process* of play and its emotional impact are more important than the outcome [54]. Collaborative scenarios are therefore artificially created to provide enjoyment rather than arising organically from the shared task or goal. Current frameworks and theories of coordination (e.g., Malone and

Crowston [40] or Eccles [19]) are either not well matched to the resources and actions in video games, or are designed for slower collaborative processes rather than real-time activity. Similarly, current tools for diagramming coordination (e.g., Gantt or PERT charts) do not adequately specify the elements present in game situations.

1.2 Motivation

Designing a video game is a complicated process as games are artificial objects made by humans and designing players' interactions makes the process harder to understand [4]: "The work of ludologists and professional game designers does provide solid knowledge about game design but does not necessarily use a structured approach based on the design of interaction [4, p33]". Although the area of playful collaborative application has been investigated frequently by CSCW¹, collaborative games have received an insufficient consideration [6] and the current frameworks and theories do not mesh with the realities of video game design.

Being able to exercise control over the coordination elements that make up a multi-player game is an important part of game design, but it is currently difficult to describe, analyze, evaluate, or design coordination scenarios in games. As a result, the design of coordination becomes an ad-hoc process, and designers may lose track of the details and complexities within their game's coordination scenarios. Although playtesting can eventually uncover most problems in the coordination design, it would be useful to have a way to discuss and analyze coordination at earlier stages in the evolution of a multi-player game, as well as a common vocabulary and notation for analyzing coordination errors discovered during playtesting. A game-specific

¹ Computer-Supported Cooperative Work

coordination framework designed to meet the game realities will facilitates the complex process of player coordination design.

1.3 Solution

To address these limitations and provide improved understanding of, and control over, coordination design in video games, I developed the PLATO framework, named for the five game elements that it encompasses: Players, Locations, Actions, Time, and Objects. PLATO provides two main constructs to designers: a set of core concepts and a vocabulary for discussing, describing, and analyzing coordination scenarios; and a diagram notation for visualizing coordinated activities. The understanding provided by the PLATO framework allows designers to better control the coordination requirements in their games, and adjust the type and difficulty of shared activities.

1.4 Evaluation

By introducing the PLATO framework, I show how my ideas can be used by designers and, demonstrate the expressive and descriptive power of the framework through a series of worked examples and a case study of coordination scenarios in Portal 2's co-op mode. Finally I introduce Stealth Hacker, a mixed reality game, and illustrate the power of the framework in coordination design for real game scenarios. These case studies show how the PLATO framework facilitates the player-coordination design process in a more understandable and descriptive manner, and the vocabulary, methodology and diagram notation provided by the framework is evaluated in different real game scenarios.

1.5 Contributions

I make three main contributions: I extend the ideas introduced by earlier theories of coordination to the domain of real-time multi-player games; I provide a new set of concepts that can be used to describe and characterize multi-player interactions; and I provide tools that can help designers produce better multi-player games through better control over coordination.

1.6 Outline of Thesis

Game design is an iterative process and the framework of PLATO can contribute to different points within the process. After a review of relevant literature in Chapter 2, I explore different situations where PLATO can aid designers with the process of coordination design. In Chapter 3 and 4, I introduce the details of the PLATO framework and then investigate player coordination in a sub-scenario of the Commandos game and in a case study of the game Portal, illustrating the approach where PLATO contributes to the last iterations of the game design after beta-testing. Chapter 5 explores PLATO as a tool that is used in earlier design phases and before playtesting. In this Chapter, I introduce a mixed-reality game called Stealth Hacker and illustrate how the framework can be used in the coordination design process of this game. Chapter 7 provides discussion and future work, and concludes the thesis.

CHAPTER 2

RELATED LITERATURE

The review of the previous work is approached through eight different areas. I start with coordination as a general concept and review prior coordination frameworks. Next, I investigate awareness as a key coordination mechanism. In the third section, I explore how current digital multiplayer games support player coordination. Fourth, a review of the achievements of researchers in kinesiology in the real-time high speed coordination domain is presented. The way researchers formulate the design of multiplayer games and user interaction as design patterns is explored in the sixth section. In addition, the role of multiplayer games and collaborative learning in educational systems is investigated in this section. Finally, the two last sections consist of social interaction design and mobile mixed reality games.

2.1 Coordination

Coordination – the management of dependencies in the actions of two or more people [40] – is a part of most shared activities, and researchers in several fields have studied group coordination (e.g., [14, 23]). Researchers have looked at coordination in contexts as diverse as ship navigation [33], surgery [51, 59], command and control [28], and shared-workspace groupware [24]. These studies show that group work has numerous episodes of coordination both at large time scales where planning and division of labour are the primary activities (e.g., industrial or manufacturing processes [55]), and at the scale of individual actions such as passing a tool from one person to another, or timing an interruption to coincide with a break in another person’s activity (e.g., [24, 29]).

There are several ways that researchers have considered coordination in prior research. One commonly cited theoretical approach to coordination is that of Malone and Crowston [40], who decompose collaborative activity into four constituent elements: Goals, Activities, Actors and Interdependencies: “Coordination means the act of working together harmoniously...Thus there must be one or more actors, performing some activities which are directed toward some ends. In what follows, I will sometimes refer to the ends toward which the activities are directed as goals. By using the word "harmoniously," the definition implies that the activities are not independent...we will refer to these goal-relevant relationships between the activities as interdependencies [40, p4].”

Interdependency among the activities creates requirements for coordination, and this theory identifies three kinds of interdependency: prerequisites, shared resources, and simultaneity. This framework was originally focused on large-scale work practices such as manufacturing, but has since been applied in other contexts as well.

Many researchers examined coordination in terms of the underlying mechanisms and explored the main factors to coordinate people such as turn taking [13], when performing a task requires taking turn by the actors, planning [55] when actors are assigned a division of the task, verbal communication (e.g., [14, 30]), roles (e.g., [52]) which are different and interdependent, Rules or structure and awareness (e.g., [25]).

Tatar et al. addressed coordination as a key to sociality by providing people’s movements and actions within a common context [62] and coordination is referred as essential aspect of cooperation in [52]. A multiplayer game does not necessarily need to be a pure collaborative game to leverage collaboration as players may collaborate with some teammates while competing with other players. Chanel et al. illustrate this difference in the players’ experience via

physiological compliance² in which the more intensity in players' interactions results in a higher physiological compliance [8].

2.2 Game Theory

Games have always been an important concept for coordination and have been studied by a variety of different disciplines such as education, politics, economics, psychology, and philosophy. Game theory is "the study of mathematical models of conflict and cooperation between intelligent rational decision-makers" [44]. It was introduced by John von Neumann with the idea of two-person zero-sum mixed-strategies equilibria [67]. In a competition, one's achievement equals the other's loss and the sum of utilities is zero [44]. Prisoner's dilemma [67] explains how two individuals defect despite achieving more by cooperating.

In a classic example of the game, a police arrests two suspects (P_A and P_B) of committing a crime and because of insufficient evidence, the police separates them into two different isolated cells. The prisoners know if they both cooperate and stay silent, both get a two-year sentence. If one cooperates and the other betrays, the betrayer goes free and the other gets a sentence of ten years. Finally both prisoners decide to increase their own utility by betrayal and as a result, each gets a five years sentence. Table 2-1 shows the payoff matrix for the two prisoners. The first number of each cell indicates the years of sentence for the prisoner in the same row and the second number shows the years of sentence for the prisoner in the same column.

² The correlation between the physiological signals of the dyad members [8].

Table 2-1. Prisoners' dilemma payoff matrix.³

	P _B Cooperates	P _B Defects
P _A Cooperates	2,2	10,0
P _A Defects	0,10	5,5

The iterative version of the prisoner's dilemma which is played in several rounds is a more a cooperative game as players are able to trace the payoffs and coordinate accordingly. This coordination can also take place without the existence of awareness in the minimal social situation [11].

A classic example of a coordination game is called Stag Hunt. In this game, players need to coordinate to maximize their utilities and the payoffs are highly dependent on the way they coordinate. The game is based on a story of two hunters. There are two rabbits which can be hunted by any of the hunters; also a stag with the greater amount of meat but players have to coordinate to catch the stag. The payoff matrix is shown in Table 2-2. If players are able to coordinate the hunting of the stag they will achieve the highest utility (5 each). However, if they do not coordinate, a player may hunt both of the rabbits and the other player will starve. The theory reveals that players in the best case coordinate on hunting the stag which maximize everyone's utilities and failing to coordinate will lead players in coordinating to hunt the rabbits with the less amount of payoff (2, 2).

Table 2-2. Stag hunt payoff matrix⁴

	P _B , Stag	P _B , Rabbit
P _A , Stag	5,5	0,4
P _A , Rabbit	4,0	2,2

In a cooperative game, players may still follow different goals and have different payoffs but in a collaborative game played as a team, players pursue the same goals and share the

³ Source: Wikipedia, The Free Encyclopedia. Wikimedia Foundation, Inc. 22 Jan 2012.

⁴ Source: Wikipedia, The Free Encyclopedia. Wikimedia Foundation, Inc. 22 Jan 2012.

outcomes [50]. The sophisticated process of modeling player and team interaction of such these games has made the area of collaborative games less interesting for game theorists [50].

2.3 Coordination and Awareness

Another area of research considers awareness support as a necessary tool for enabling coordination, particularly in distributed groupware [14]. There has been considerable attention about awareness in CSCW during recent years. As a general approach, awareness is a real-time implicit understanding of dynamic environments which is gained by user's interaction and exploration. The two key points in maintaining awareness is extraction of necessary information and providing a proper presentation of that data for group members [23]. Gutwin and Greenberg [24] studied real-time distributed groupware and highlighted the important role of awareness in collaboration. They defined the concept of workspace awareness as “the up-to-the-moment understanding of another person's interaction with the shared workspace” which facilitates coordination of group members and simplifies their communications [26]. The latter concept is also similar for awareness of distributed players of an online multiplayer game by providing players with the answer to questions such as “where are the other players”, “what are they doing?”, “what are they going to do next?”.

Researchers have looked at awareness requirements at several time scales, including traces of activity over long time periods [25] and real-time awareness of immediate actions [61]. They studied the role of awareness information on teamwork and the way people coordinate their actions, and examined different ways of providing this information in distributed collaborative systems.

Despite this work on real-time awareness, there is little work in CSCW on coordination requirements when teamwork happens quickly and when there is insufficient time to coordinate verbally. One exception is Hutchins [33] which investigates how complex navigation tasks are managed by teamwork and explains the key role of tools and interpersonal interactions in managing the interdependencies achieved by labour division. In this context, the time pressure (and severe consequences of failure) forces the activity into a highly regimented structure, where coordination is simplified through strict division of labour and adherence to procedure.

Heath and Luff [28] in a study of the London underground control room explained the complex mechanisms underlying the collaboration between the Line Controller and Divisional Information Assistant and the use of overhearing and monitoring and interaction via obscured actions to coordinate. Gutwin and Greenberg [25] explained the need of supporting workspace awareness information for users of real-time groupware. Dourish and Bellotti [14] in a study of shared editors as CSCW systems highlighted the role of providing passive information for the users in supporting their dynamic coordination.

2.4 Coordination in Digital Multiplayer Games

Studies of coordination in multiplayer digital games have also been performed. Researchers have considered coordination in real-time distributed games [14]. Tatar et al. [62] studied player's coordinated activities in playground games and explained several reasons for the importance of coordination in such environments: They “1. Teach us about the processes inherent in human coordination 2. Provide a model of desirable coordinative possibilities and 3. Act as a design framework from which to explore the relationship between game and game play.” She explored several simple multiplayer games based on the three concepts of rules, roles

and turn taking and showed that with an appropriate design, these systems can provide players with an experience of plurality, ‘appropriability’ and ‘acompetition.’

Researchers have also focused on awareness information that is necessary to support fast coordination of activity. Nova et al. [45] investigated collaboration in multiplayer games and examined the effects of awareness tools on mutual modeling and teamwork. He showed that players who were taking advantage of awareness tools were acting more successful in collaborative play and achieved better results. Nova in [46] focused on Quake-like games and used the framework of workspace awareness [25] to study awareness support in digital games. He explained that “location, presence, identity, action and event history” are the major in-game awareness-support tools. Tang et al. [60] also explored this area by studying two different multiplayer online games and discovered that the need for in-game awareness support is similar across different game genres and that players have similar expectations of awareness and interaction supports in playing different games.

Other studies have looked at the coordination problems caused by network delays, showing that delays of even 100ms can have substantial effects on people’s ability to coordinate tightly-coupled actions [24]. Pinelle et al. [48] in a study of multiplayer network games presented a set of heuristic to measure the game usability that can affect the process of game design before playtesting. The research explains the heuristics in ten different factors of simple session management, flexible matchmaking, appropriate communication tools, support coordination, providing meaningful awareness information, identifiable avatars, protected training for players, social interaction, lower delay and management of bad behaviors.

2.5 Coordination in Kinesiology

Researchers in kinesiology have more frequently examined the area of real-time coordination in their explorations of how sports teams function. For example, Eccles and Tanenbaum [18] define coordination as “integrating the operations of the team in a timely way to form a composition of operations that achieves satisfactory performance” and look at several aspects of coordination in team sports. These studies identify that coordination becomes difficult when time pressure is high [18], and performance is strongly affected by a team’s ability to use pre-determined ‘plays’ that have been developed and learned in practice [19].

Eccles presents a framework that identifies communication, assumptions, planning, and reflection as critical elements in shared activity, and covers pre-activity, during-activity, and post-activity phases [19]. He explains the process of coordination takes place when the correct type of actions happen at the correct times and locations and refers to interdependency and interdependent contributions of team members as the key to success of teamwork. This interdependency can be achieved when the divisions of labor that is assigned to actors has been divided differentially to connect activities of members in terms of type, location and timing.

2.6 Cooperative Game Design Patterns

Providing support for players’ interactions and collective actions has become a crucial factor in multiplayer games’ design process, and research into a variety of tools and frameworks to facilitate the process of game design have been reported. Bjork and Holopainen [4] explained supporting in-game social interaction makes the process of game design more complicated and raises the need of interaction designers to contribute with the design process. They present a

comprehensive activity-based framework as patterns for game design. With the focus on gameplay, their research investigates the area of digital games from different perspectives and decomposes the game mechanics to the main underlying design elements. Their framework addresses the lack of terminology in the design of gameplay and provides game designers with a set of possible design choices as well as a language to describe digital games based on players' activities. They represent the game design patterns to support players' social interaction in four different approaches of competition, collaboration, group activities and stimulated social interaction.

Some researchers addressed the insufficient player interaction supports in games and studied the specific design requirements of multiplayer games for coordinating players and leveraging collective actions. Zagal et al. [70] refers to the solitary approach of particular digital games in contrast with the collectivity nature of games. Their research raises the need of having collective actions and interactivity as the most important factors in games. This provides more interesting experiences for players by playing with or against other human players instead of artificial intelligence. The model defines the main concept of a game in two major parts; rules and goals which explain the players' ability and the objective of the game, and props⁵ and tools that make game playable. It refers to the six elements of social interaction, cooperation and competition, synchronicity, coordination, prop and tool dependence, and meta-gaming as the main characteristics of multiplayer games.

Many other researchers also studied methods to formulate cooperative game design by investigating the general specification of these games. Rocha et al. [52] studied multiplayer game design patterns and mechanics underlying the cooperative games. Their research explains these

⁵ A prop is a purely decorative item in the game world [70].

patterns as seven key factors of complementarity roles, synergic abilities, interpersonal activities, shared goals, interlaced goals, internal team rules. El-Nasr et al. [56], in a study of cooperative games, adds a new set of design patterns to the latter achievements [52] as well as a metric set called Cooperative Performance Metrics (CPM) to evaluate cooperation. Camera settings, using shared objects, shared puzzles, shared characters, vocalization and limited resources are additional game design patterns for cooperative games. She evaluated the underlying cooperation based on excitement, successful strategies, assistance and global strategies that took place among the players while cooperating within a multiplayer game.

Zagal et al. [69], in a similar approach to Jochen and Hsi [50], investigated the area of collaborative board games and highlight four different key concepts and three pitfalls of the underlying collaboration design. They address two major problems within the design of cooperative games: “free riding” and “back stabbing”. They suggest that designers can take advantage of tension in players’ choosing between their own utilities and team utilities; players should be able to make a decision without the team approval by relying on the ability of tracing actions’ consequences. Assigning specific responsibilities to players can encourage making selfless decisions. Their research indicates that designers need to use effective mechanisms to avoid solitary play and they need to assign meaningful outcomes to players’ actions to make the game more enjoyable. In addition, the game should be designed in a flexible way in which tasks can be completed in different ways each time the game is played.

2.7 Collaborative Learning

Games are an important research topic for collaborative learning and researchers have frequently investigated the area of collaborative games. Collaboration is an effective way to

leverage motivation [12] and coordinating students to collaborate on a certain topic has an effective role in student's learning in many instances. Jochen and Hsi [50] have studied the area of collaborative games with a focus on learning potentials. By highlighting the tendency of human beings toward competition more than collaboration, they highlight the lack of useful collaborative models for students. Their research investigates the key concepts of designing game scenarios which actuate the players to coordinate and play as a team, and presents a framework for analyzing collaboration by using game theory.

Dickey, in [12], studied the massively multiplayer online role-playing games (MMORPGs) as a source of game-based learning environments and the necessary supports for students' intrinsic motivation. She decomposed motivation into five main factors of choice, control, collaboration, challenge and achievement and explored the related design requirements from the two perspectives of character design and narrative environment. This research highlights the role of collaboration in motivation by creating peer role models for players and coordinating them.

Malone and Lepper in [41] studied the methods of providing intrinsic motivation for learning and highlighted its role in the way students learned new concepts even if they are extrinsically involved in a certain activity. They explained an intrinsically motivating activity as "if people engage in it for its own sake, rather than in order to receive some external reward or avoid some external punishments [41, p3]". Competition, cooperation and cognition as interpersonal motivations were some types of intrinsic motivation. Competition and cooperation are explained as two meaningful parallel concepts when outcome of people's actions are assigned to utilities in achieving a certain goal(s). Coordinating students by using independent activities need exogenous techniques while dependent activities provides better results via

endogenous cooperation or competition. For the first case, the motivation can be achieved by mixing the players' scores though it may not be strong enough while the latter one brings endogenous cooperation by dependent actions.

2.8 Interaction Visualization

Several researchers have looked at tracing and characterizing interaction and coordination in collaborative learning systems. Prior work has highlighted the role of activity visualization in quantifying, analyzing and designing coordination. Some researchers have investigated collaborative activities by visualizing and describing action logs using diagrams [e.g., 63, 47] or regular expression [47]. They have explored different visualization methods such as dependency graphs [58] to study interactional construction of meaning between two collaborators.

Harel [27] used “higraphs” to create diagrams in the style of Finite-State Machine (FSM). In this approach, he used hyper-edges to attach different nodes of a visualized set (Figure 2-1). The represented edges connect graph nodes in a certain direction and can hold diamond-shaped labels describing the relationships between the connected nodes. This FSM-style visualization can represent a system at different levels of abstraction by hiding or showing the relations between nodes at different levels of nested states.

2.9 Social interaction design

Games, like sports, are an interesting case for coordination because the difficulties of managing shared work can actually be an important part of the fun of the activity – and researchers have looked at the explicit connections between coordination and sociality (e.g., [60, 46]). In many of the work domains studied in previous research, management of interdependencies is required to achieve productivity, but in multiplayer games, the process of

play and its emotional impact are more important than the outcome [54]. Collaborative episodes are therefore artificially created to provide enjoyment rather than arising organically from the shared task or goal. Computer games have benefited from the added enjoyment of collaborative or competitive play [60].

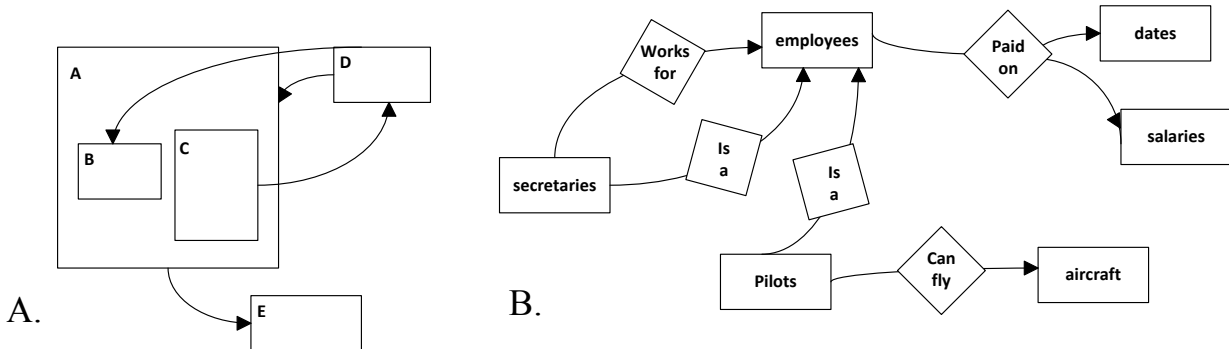


Figure 2-1. “higraphs” as a methodology of diagramming in the form of finite-state machine [27]. Figure (A) represents internal and external nodes within a system and their relations. Figure (b) demonstrates the use of hyper-edges for illustrating different types of relations between nodes.

Social presence for supporting sociality in digital games has been an important research topic and a variety of research has been performed in this area to support player interaction and coordination. Maninen in [42] investigates the underling social interaction forms in multiplayer games from players’ perceivable actions as a tool to evaluate the level of interaction supports in games. Guillaume et al. [8] use another method for this measurement by taking advantage of physiological compliance. This study of players playing cooperative and competitive multiplayer games results in higher physiological compliance by increasing in the level of interactions.

The popularity of online multiplayer games has affected the arena of solitary digital games. “Killing a monster” and gaining experience points, is not the only goal of playing digital

games anymore and socialization into the game community has become a key factor of play [15]. Ducheneaut and Moore in [15] studied the social interactions within EverQuest [35] and explained the underlying interactions in four different categories:

1. Self-organization among players: Persuading players to create and maintain their own team and act as a community member.
2. Instrumental coordination: Making players to coordinate their actions and perform teamwork to achieve the goal.
3. In-game sociability: Referring to player's informal communication which increases the team coherence.
4. Helping behavior: Leverage information sharing and assistance within a collaborative team.

Vogiazou and Eisenstadt [65] examined the role of symbolic presence and good visualization on spontaneous group self-organization and coordination. They created a 2D multiplayer game application with simple movements and an abstract and non-verbal communication support in which players were identified by colored circles (bumper cars) in two competitive teams. Despite the simplicity of the game, players succeeded in coordinating their actions and following a series of spontaneous harmonic actions such as rotation and changing color by taking advantage of visual clues. The study reports six different observed behaviors: subverting the game, leading, expressive group performance, goal-based teamwork, self-organization and drawing attentions.

Joint actions as the key determinant in supporting player social interaction and player coordination have been frequently studied. Chen and Duh [10] studied the underlying social interaction forms in World of Warcraft as well as the key factors that can influence these

interaction forms. This study states that “instrumental joint acts” is an in-game factor that shapes the social interaction forms. This takes place when players as “social actors” perform common collective actions such as trading exchanges. “Rules of conducts” was another in-game factor that influenced the social interaction forms within World of Warcraft. Civil-legal rules, looting rules, guild rules and socializing norms were referred as the examples of the rules of conduct in the game.

Stimulated and natural interactions among the players can also become available via the specification of different locations in the game world. Ducheneaut et al. [16] studied the social dynamics of Star Wars Galaxies and explored the way the game design supports in-game social interactions. The research shows that the effective use of space in different locations successfully engages players with common activities. For example, the purposeful design of locations made areas like the “cantina” and “starport” act effectively in gathering a large numbers of players at the same time. The concept of space is regarded as the primary interdependency factor, yet the game suffered from shortcomings such as the lack of awareness support.

Ducheneaut et al. in another study [17] using longitudinal data of play time, grouping and guilds, explored the game as a social environment with the emphasis on game design more than social aspects. Despite the consideration of the numerous players who play alone in community and the lower popularity of joint actions, participants referred to the social factor as their motivation of playing a massive multiplayer online role-playing game (MMORPG). The study reveals that although some players play individually, they are more interested in playing in this community rather than equivalent single player games due to the audience the game provides, a sense of social presence and spectacle.

Communication is an effective coordination mechanism and a key factor for supporting social interaction has also been studied. Huffaker et al. [30] considered communication skills as one of the requirements of an expert player and studied the social interactions of experts in EverQuestII. In this study, expertise was composed of two different factors: level of achievements and efficiency of play. By comparing the level of players' communication, they showed that achievers took advantage of more social interactions in terms of communication with other players but performance players' interactions were the same as other players. They explained the fewer tendencies can be a result of the time-consuming process of communication for the players and also the fact that experts can be play independent to others advice.

Brown and Bell [6] in a different approach, studied the social aspect of a massively multiplayer game called "There" as a collaborative virtual environment (CVE). They highlight the importance of social actions and stated that it can play a more important role than some other factors such as social presence and social ties. Interacting with strangers is encouraged in the environment of "There" and shared activity and interaction around objects are the important factors of social supports. The game takes advantage of a flexible open environment that allows the users to choose their desired social activities. Taking advantage of shared objects, integrating overlapping chat with the game environment as well as providing safe communication atmosphere regarded as the strength of the game.

Some researchers have identified that designing for coordination can be a problem in games. Tatar et al. [62] indicates shortcomings in current methods for designing games, and looked at design ideas for coordination in playground games. As a result, game designers must be able to balance the coordination requirements such that an activity is possible, but difficult

enough to be challenging, and fun for the group. However, structured capacities to design or even discuss these collaborative game elements have been lacking.

2.10 Mobile Mixed Reality (MMR) Games

In recent years, starting from games like Pirates! [5], a variety of MMR games have appeared – a genre that is played in real and virtual playgrounds simultaneously. Can You See Me Now? [2] was a successful implementation of this type of game with an approach to leverage teamwork in which a team of “Runners” communicate to chase the opposite team and play an elaborate game of team tag in the virtual and real world simultaneously. Epidemic Menace [37], Treasure [7] and Uncle Roy All Around You [3] are other mixed reality games requiring various levels of coordination and teamwork. CitiTag [66] was another MMR implementation that emphasized the concept of presence, aimed to support and fortify group work and intentional communication. In this specific game genre which highly leverages player interaction, coordination plays a crucial role among the team of players in both virtual and real playgrounds. Inderbitzin et al. [34] studied the behavior of two teams of players collaborating and competing in a mixed reality football game. They showed that players with coordinated movements left fewer gaps in between and decreased the chance of losing points.

Despite the emphasis on the key role of player collaboration and social interaction in the previous literature, and the investigation of coordination on a variety of domains, player coordination is currently the domain that suffers of insufficient theories and methodologies that match well with the realities of multiplayer games. In the next chapter, I introduce PLATO as a coordination framework designed specifically for games, and explore the way it addresses the previously existing player coordination issues and shortcomings.

CHAPTER 3

PLATO: A FRAMEWORK OF COORDINATION IN GAMES

The PLATO framework is aimed at providing improved understanding of, and control over, coordination design in video games. It is named for the five coordination elements that it encompasses: Players, Locations, Actions, Time and Objects. PLATO enables designers control over the way players interact with each other and with the game environment. PLATO provides a vocabulary and terminology for the design team to communicate and describe collaborative scenarios as well as a set of tools and criteria for formulating and designing coordination. In other words, PLATO provides game designers with a tool for conceptualizing, describing, analyzing, and designing specific coordination episodes.

To specify the details of a coordination episode, the framework must identify relevant conceptual entities that can play a role in the coordination. Malone and Crowston's theory [40] employs the entities which are valuable but insufficient to capture the subtleties of coordination in digital games. I have adapted Malone and Crowston's ideas to better suit the specific needs of multi-player game design. In PLATO, I assume that the goal of the episode is determined by the game designer, and that specific types of interdependencies should be explicitly modeled in the framework. Therefore, I build PLATO on five concepts that can engage coordination requirements: *Players*, *Locations*, *Actions*, *Time*, and *Objects*.

A coordination episode in a game is a situation where players must coordinate, either to achieve a goal or to improve an aspect of their performance. This collaborative scenario consists of several atomic actions performed by players. Eccles [19] refers to the atomic actions as elementary actions, achieved by a differential division of labor: "task is broken into elementary

actions, defined as actions that cannot be divided further without a marked reduction in task performance.” Each atomic action is shaped up of a subset of coordination elements (Players, Locations, Actions, Time, and Objects) implying that the atomic actions can be explained upon the underlying coordination elements. These coordination elements may remain consistent across a number of atomic actions in a certain coordination episode. I refer to these inter-activity elements as interdependency factors which connect and weave several atomic actions in a certain episode.

3.1 PLATO Part 1: Players

A multi-player coordination episode depends on players carrying out actions in the shared virtual space of the game. Obviously, the number and type of players (or their characteristics) is a central component of the coordination episode. Players are part of every PLATO analysis, as they are the agents in the game world. There are two primary aspects to specifying a player element: which game entities can act as players, and what degree of specificity is needed to characterize the coordination episode.

3.1.1 What game entities can act as a player?

Players are generally avatars or characters that can be directly controlled by people. For example, the player is obvious in first-person games (e.g., Enemy Territory [20]) or third-person games (e.g., Tomb Raider [64] or Super Mario Bros [43]) where the human directly controls a single avatar explicitly represented in the game world. The embodiment of the avatars is highly tied to people’s achievements in game and is their only medium to influence the game state [4].

There are several possible variations on this general case. In some cases the player is represented by a vehicle (e.g., a car in a racing game), and in strategy games (e.g., Starcraft), the player can be the invisible but omniscient commander of the units in the world, or could be an individual unit in the game. Some games (e.g., team sports games or RPGs) may have a number of playable avatars. For example in FIFA Soccer [22], one player among the eleven players of a team become avatar and the rest is controlled by AI.

For our purposes, *players* can be any of the avatars or units that could be directly controlled by the human player, including those who are temporarily under the control of the game's AI. Our definition uses an earlier organization of player types [44] into three tiers: main characters which are under player control, characters that assist the main character as henchmen or assistants, and characters that passively join parties to provide limited aid and advice.

3.1.2 How are players specified?

There are several ways that game requirements could constrain a player's identity, role, or capabilities in a coordination episode.

- *Specific role or characteristic.* The player must be of a particular type (e.g., character class in World of Warcraft [68]), have certain attributes (e.g., be on a particular team) or have particular abilities (e.g., be able to defuse bombs).
- *Identity.* The episode can only be carried out by a particular character identity, or even a particular human-player identity (e.g., only the player who instantiated a quest instance can participate in dialog).
- *No constraint.* Any player can fulfill the requirement.

Table 3-1 illustrates the summary of player specifications.

Table 3-1. Player specification summary

Player specifications	Description
Specific role or characteristic	Particular type, attributes or abilities
Identity	Particular character identity
No constraint	Any player

3.2 PLATO Part 2: Locations

A game world is “The environment in which the gameplay or parts of the gameplay takes place is determined by the spatial relationships of the game elements” ([4] p.55). It can consist of accessible and inaccessible areas. The accessible area for a specific player may not necessarily be accessible for another player type [4] which can be used as a feature to increase interdependency in the game scenario.

In PLATO, Locations are points or areas in the game space where players can be, where coordinated activities can occur, and where objects can be placed. Most game worlds have a spatial organization of some kind (e.g., a sports field in FIFA Soccer [22], rooms and passageways in Portal 2 [49], or an instanced world metaphor in World of Warcraft [68]), and this reference frame is used to specify locations when describing coordination episodes with PLATO.

3.3 PLATO Part 3: Actions

An action is any change made to the game state by a player – for example, kicking a ball in a soccer game, shooting a rifle in an first-person shooter game, or casting a spell in World of Warcraft: “Actions are the means through which the player can make changes to the game state”

and therefore the physical act of pressing the joystick button is not assumed as an action by itself ([4] p.20).

Actions are involved in most coordination episodes, although it is possible to construct episodes that do not require actions – for example, a territory-based game might require that three different players simply exist in three particular locations simultaneously, without carrying out any particular actions at those locations. Actions are, of course, dependent on the nature and domain of the game.

In some cases, activities that are closely related to other PLATO elements (particularly locations and objects) are not represented in descriptions of the episode. For example, “move to location X” or “hold object Y” are usually implied by constraints in the location or object elements. Similarly, actions can imply an object, and in these cases it is often not necessary to explicitly state the object as part of the coordination. For example, an action such as “fire gun” implies that the player has a particular object (i.e., a gun), and the object does not need to be stated. These flexibilities in the way scenarios are specified underline the fact that PLATO is not intended as a formal specification tool, but rather as a resource for designers to discuss, understand, and plan coordination requirements.

3.4 PLATO Part 4: Time

Time is a critical aspect of coordination requirements in multi-player games, and temporal constraints on a shared activity are often the key component employed to make the activity challenging. Actions affect the game state and influence subsequent actions. Thereby, an illustration of ordering the actions based on time can describe the game events ([4] p.27).

There are two main kinds of time requirements in PLATO: ordering and clocking.

- *Ordering (event-based activities)*. Some activities must be carried out in a particular order (e.g., in Enemy Territory, the bridge must be built before the tank can be moved forward). One sub-type of ordering is *simultaneity*, where actions must occur at the same time.
- *Clocking (time-based activities)*. Some activities must happen at particular times, or during particular time periods, or before a particular time limit. Two sub-types of clocked coordination are *soft clocking*, where there is some leeway on the timing of the activity (e.g., within ten seconds of time T), or *hard clocking*, where the temporal requirements are strict.

From these two main concepts, I define several specific types of temporal coordination constraints that are used in multi-player games. I note that this is a main area in which PLATO goes beyond previous coordination frameworks: where two temporal classifications were sufficient in Malone and Crowston's framework, I define numerous types that better capture the richness of process interaction inherent in games. In the list below, I use the term 'operation' to imply either an action by a player, or an event such as a player occupying a location.

3.4.1 Temporal classifications

- *Mutual Order*: A series of operations must be performed in a specific sequence.
 - *Example*: Atlas in Portal⁶ creates a portal for P-Body. P-Body uses the portal and opens a door for Atlas.
- *Clocked Order*: Operations must be completed mutually by each actor before a timer expires.

⁶ Portal 2 Co-op mode, chapter 1, course 1

- *Example:* The Spy in Commandos must kill a soldier and the Green Beret must take the body away before the patrol approaches.
- *Individual Order:* A player must perform a series of operations to allow other players to proceed.
 - *Example:* P-body in Portal⁷ shoots to another target to let both players begin to free fall, before touching the ground creates another portal on the ground while Atlas is idle.
- *Soft-Simultaneity:* Several operations must be completed within a short time period (Tss).
 - *Example:* When allies aim to conquer an enemy target in Age of Empires, they must approach to the enemy location within a soft-simultaneous time span.
- *Hard-Simultaneity:* A series of operations must be accomplished with negligible delay (Ts).
 - Atlas and P-body in Portal⁸ need to push two handles simultaneously to release an edgeless cube.
- *Parallel Activity:* Players must complete a series of operations independently, but the final objective cannot be completed until all have finished.
 - *Example:* Atlas and P-body in Portal⁹ each needs to push different series of buttons in parallel.
- *Clocked Parallel:* Parallel activity with a time limit.
 - *Example:* Chris and Claire in Resident Evil 5 each must shoot a specific number of zombies when they are invaded.

⁷ Portal 2 Co-op mode, chapter 4, course 8

⁸ Portal 2 Co-op mode, chapter 1, course 6

⁹ Portal 2 Co-op mode, chapter 1, course 2

- *Order-in-soft-simultaneity*: A specific ordered set of interdependent operations, performed in a limited time.
 - *Example*: In Counter Strike, one player might have to throw a grenade into a room at the same moment that another player kicks down the door.
- *Bound*: An operation by one agent must start prior to and end after another player's operation.
 - *Example*: In Portal¹⁰, Atlas needs to keep the view open for P-body to make him able to target the turrets by reflecting the Thermal Discouragement Beam.

3.5 PLATO Part 5: Objects

The final element in PLATO is the Object – a non-playable game element that is required for the coordination episode. In another definition, objects “describe different types of game elements that can be manipulated by players or are game elements through which players manipulate the game state” ([4] p.70). As an example, an economic transaction in games can be referred as a very common activity using objects in social scenarios and player coordination increases the social interaction that takes place on exchanging the objects [6].

3.5.1 Object types

Objects can have different specifications. PLATO considers three kinds of objects:

- The first group comprises passive objects which only have a state, and no actions (e.g., a key for opening a locked door or a flag in a capture the flag scenario).

¹⁰ Portal 2 Co-op mode, chapter 1, course 3

- The second group is active objects, which have a limited set of contingent actions: for example, a mine in a multiplayer shooter is an active object because it can explode when an enemy is nearby (and thus can carry out an action as a proxy for a player).
- The third group is AI objects which are equipped with a limited set of AI functionality. They are often designed under the point and click control in video games; most units would be AI objects.

The distinction between Players and AI objects can be subtle, and can change during a single scenario. Due to the close gap between the boundaries of AI objects and subsets of Players in some aspects, it may become possible for some entities to move from one category to another by changing the player's frame of reference. For example, in the game *Mercenaries*, when the player battles an enemy tank it is an AI object. If the player captures the tank, however, it becomes a temporary player avatar. Unlike the usual scenarios in Real-Time Strategy (RTS) games in which the player commands number of different units, in some scenarios, the player must complete the objective with a limited number of units; for instance by using a single tank. In this case, the tank is more an avatar than an AI object.

Again, these subtleties can be dealt with by the designer; while these definitions are the widely appropriate, it is possible for designers to use alternate definitions that are best suited to their game. Changes in the distinction between Player and Object do not affect the overall methodology or utility of the framework. Table 3-2 illustrates the summary of Objects specifications.

Table 3-2. Different types of objects and their specifications

Object types	Specification	Example
Passive	They have a state and no actions	A key for opening a locked door
Active	They have a limited set of contingent actions	A mine in a multiplayer shooter
AI	They have a limited set of AI functionalities	Most units in RTS games

3.6 Goals and Interdependencies

In the PLATO framework, I define the term *general goal* as the abstract objective of the collaborative episode that can be used as the title of the episode, and *specific goal* as a clear achievable goal of the coordination episode. A specific goal usually implies the task that will be defined based on the specification of the episode and assigned within the existing roles in the game. The task defines the set of actions that are interdependent through the coordination elements and performed by the players. I refer to these factors that connect the actions of players as *dependency factors*. Depending on the scenario, players' activities can be interconnected by a subset of coordination elements from Player, Location, Time and Objects. Usually in the episodes that players have different roles, abilities and responsibilities, Player contributes as a dependency factor. Similarly, Objects as shared resources are often a type of interdependency. For example consider a simple scenario in which two players (P_1 , P_2) must turn two handles at the same time to open a gate. In this episode, Players (P_1 , P_2), Actions (turning the handles) and Time (simultaneity) are the coordination elements, the general goal is opening the gate, the specific goal is turning the handles simultaneously and the dependency factor is Time. In the next chapter, with the focus on analysis, I will explore the coordination elements in more details and will evaluate these concepts in different scenarios. Table 3-3 illustrates a summary of the PLATO Framework.

Table 3-3. PLATO Framework summary

PLATO Framework
Elements
Goals (general and specific)
Interdependencies

Table 3-4. Elements of the framework

Elements
Players
Actions
Location
Time
Objects

Table 3-5. Interdependencies in PLATO

Interdependencies	Specifications
Players	Players with different role and characteristics
Locations	Locations that are tied to collaborative actions
Time	Timings that interconnects the actions
Objects	Common objects and shared resources

Game design is an iterative process and the framework of PLATO can contribute to different points within the process. In Chapter 5 and by investigating a mixed-reality game I will use the framework in the coordination design process of this game and show how PLATO contributes in earlier design process before playtesting. In Chapters 3 and 4, I will illustrate the approach where PLATO contributes to the last iterations of the game design after beta-testing.

CHAPTER 4

ANALYZING COORDINATION

Being able to exercise control over the coordination elements that make up a multi-player game is an important part of game design. It is useful to have a way to discuss and analyze coordination at earlier stages in the evolution of a multi-player game, as well as a common vocabulary and notation for analyzing coordination errors and opportunities discovered during playtesting. PLATO provides a set of tools to assist game developers and researchers by providing a common language and criteria for formulating and quantifying coordination.

The first step in analyzing coordination using PLATO is to constrain the scope of analysis to a single coordination scenario. Games are complex, and will likely incorporate many intermediate activities and goals. If a discrete game goal contains one or more coordinated actions or events then it is a coordination scenario, and can be analyzed using PLATO and the associated notation.

In Eccles' work [19], atomic actions are the elementary actions achieved by a differential division of labor. Atomic analysis in PLATO describes the atomic actions based on time. Analysis begins with the assignment of a time tag to a set of activities (for example T1–T2 which implies the time range from T1 to T2, | $T1 < T2$, see Table 4-1 for details).

Table 4-1. Time elements and their diagram notations

Timing	Symbol	Sample Diagram
Mutual Order		
Clocked Order		
Parallel Activity		
Clocked Parallel		
Individual Order		
Simultaneity		
Soft-Simultaneity		
Bound		
Order in Soft-Simultaneity		

4.1 PLATO as an Analytic Tool

PLATO has utility as a set of concepts for analyzing and discussing coordination in collaborative games, but to increase the power of the framework for more practical scenarios, I have also devised a diagram notation to help designers visualize how the different coordination elements interact in a given collaborative scenario. I propose two types of charts, Player-Action-Time (PAT) and Location-Player-Time (LPT) charts. Other types are possible, but I have found

these to be the most descriptive. In PAT charts, time is on the x-axis and players on the y-axis. Annotated actions for each player are drawn as horizontal lines, and timing and collaborative relationships are drawn as a second set of lines according to the notation guide in Table 4-1. PAT charts are particularly useful for elucidating the role of parallelism and simultaneity in collaborative scenarios. LPT charts also have time on the x-axis, but have labeled locations on the y-axis, and plot players as lines moving from place to place over time. LPT charts are useful in understanding how space is used in collaborative scenarios.

To demonstrate how these graphs are created, I refer to the simple scenario in which two players must cross a dark hallway with switches at either end (Figure 4-1). Either switch can turn on the light but is spring-loaded and must be held. One player must hold the first switch while the second traverses the hallway, then the second player holds the far switch while the first player moves, as shown in Figure 4-2. The general goal of this episode is passing through the corridor, the specific goal is keeping the light on and dependency factors are Time and Location.

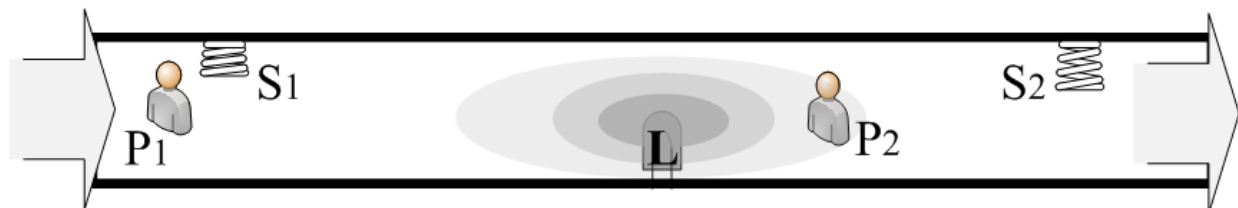


Figure 4-1. An schema of the hallway scenario. The first player (P1) is holding the first switch (S1) and is keeping the light (L) on. The second player (P2) is passing through the hallway to reach the second switch (S2).

The location diagram (Figure 4-2) clearly shows the transient-residency imposed by the collaboration in the sloped and straight lines for Players A and B. The timing diagram (Figure 4-2) shows the Bound (see Table 4-1) actions associated with the holding of a switch and

movement down a hallway. Although simple, the example shows the ease of building the diagrams and how this quantification and provided vocabulary might be employed in analysis.

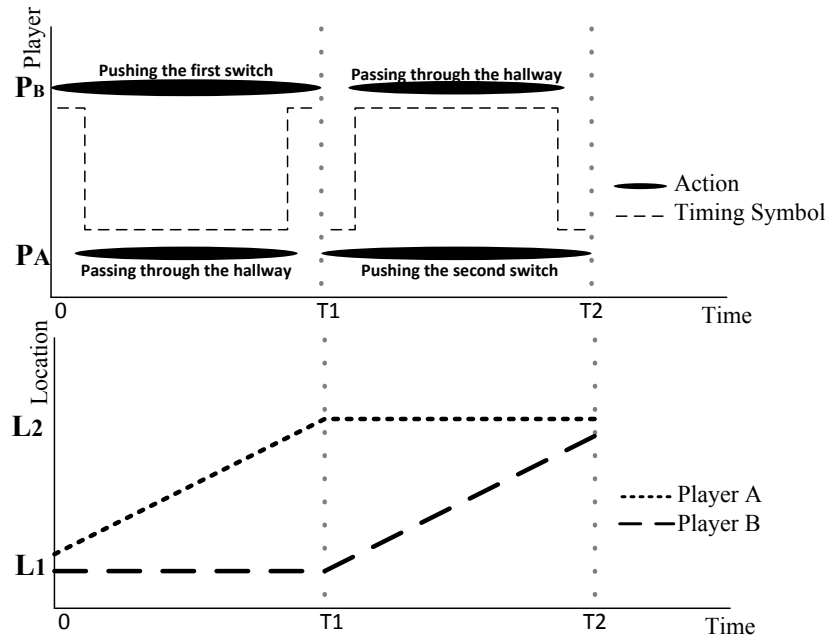


Figure 4-2. PAT (top) and LPT diagrams for the dark hallway episode. L1 represents the location where the first switch is located and L2 refers to the location of the second switch.

4.2 Example: Commandos

The Commandos game is a real-time tactical combat game [31] in which players with different roles and abilities need to plan and coordinate to achieve the game objectives. A sub-scenario of Commandos (shown in) is an example of an episode which contains a complete set of coordination factors. In this specific scenario Players should perform a number of coordinated actions to break into a sentry point without being detected. The players are composed of Driver, who carries a cigarette, Marine, equipped with a harpoon gun, and the Green Beret. To complete their objective the driver should throw the cigarette in front of the first enemy soldier. Almost simultaneously the marine should kill the second soldier with his harpoon and shortly thereafter

the Green Beret should carry the dead body away while the first soldier is still investigating the cigarette.

In this section I illustrate how PLATO can assist game designers to quantify and analyze the underlying coordination of player activities. This takes place by discovering the main coordination elements in the scenario in a component analysis process (shown in).

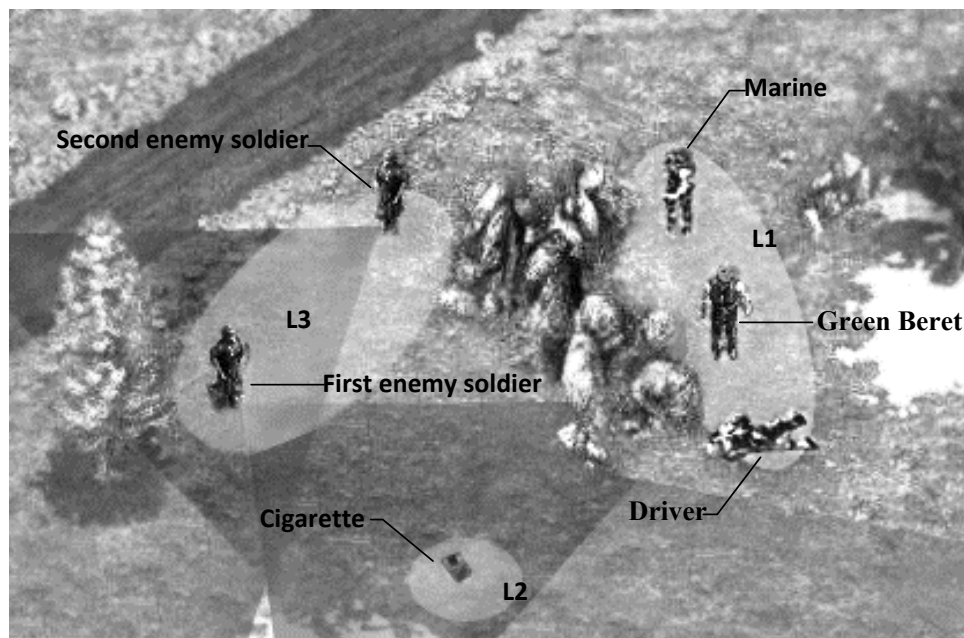


Figure 4-3. Coordination episode in Commandos, showing locations of soldiers (L3), their views (triangles), cigarette (L2) and commandos (L1) at the beginning of the episode.

The process of analysis begins with decomposing the interdependent activities into atomic actions and specifying the underlying coordination elements for each of the actions. The elements that are in common among the atomic episodes are the dependency factors, making the activities interdependent and playing key roles in player coordination. Using the PLATO diagram notation I explore the visualized data to provide a better understanding and explanation of the scenario's player coordination. I finally demonstrate how the changes made in the diagram

can affect the game balance in terms of level of coordination and difficulty. Table 4-2 illustrates the component analysis of this episode.

Table 4-2. Component Analysis of an episode in the game Commandos	
Elements	Entities in the game
Players	Green Beret, Marine, Driver
Locations	L1, L2, L3, L4, The areas under the vision range of soldiers
Actions	throwing cigarette, killing the soldier, carrying the dead body
Times	order-in-soft-simultaneity
Objects	Cigarette

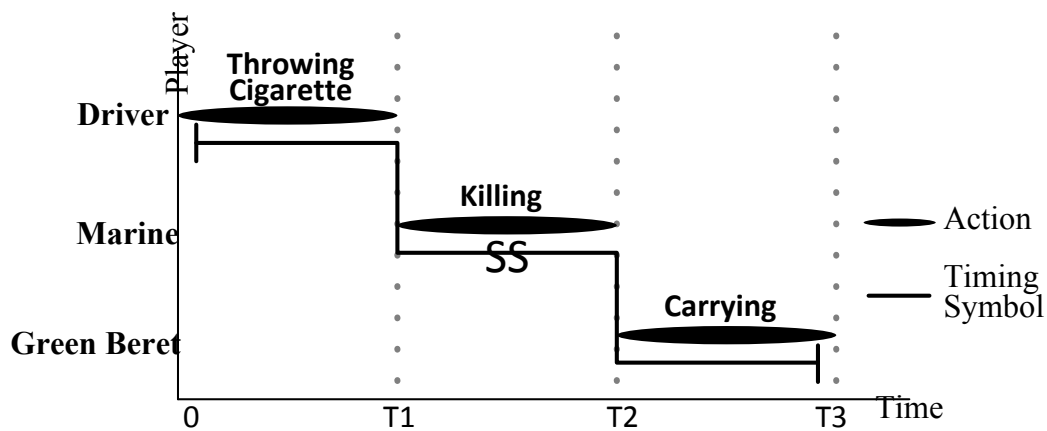


Figure 4-4. PAT diagram for Commandos sub-scenario

The analysis of this episode starts at (0-T₁): where the Driver throws the cigarette to the correct location (PALO). At (T₁-T₂) the Marine shoots the target at the L4 location (PAL). At (T₂-T₃) the Green Beret moves the dead body out of the first soldier's sight (PAL). The general goal of this sub-scenario is breaking a sentry point and specific goal is killing a protected enemy soldier. In the atomic analysis, L4 was repeated at (T₁-T₂) and (T₂-T₃) connecting these atomic actions which implies "L" as an interdependency factor. In addition, due to the different role and

abilities of the players “P” is another dependency factor. These atomic actions are connected by the “order in soft-simultaneity” timing type (see Table 4-1) introducing “T” as the third interdependency factor for this coordination episode. Thereby, dependency factors for this episode are Time, Location and Player (PLT). Figure 4-4 and Figure 4-5 illustrate PAT and LPT diagram of this episode respectively.

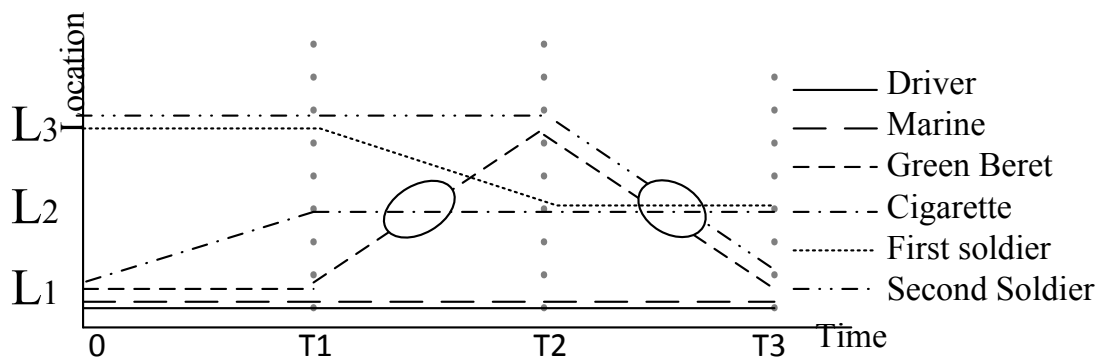


Figure 4-5. LPT diagram for Commandos episode. Locations are illustrated with different dashed lines. Circles in the diagram are used as bridges over the other lines.

4.3 Game Balance Adjustments

While PLATO serves as an analytic framework for classifying coordination activities in digital games, it more usefully serves as an analytic framework deployable at both the paper-prototyping and play-testing stages. I provide two fictitious situations based on the scenario above to illustrate how PLATO can be applied to design.

The first episode, depicted in Figure 4-6, is imagined to take place during play testing. In this episode the duration of the cigarette burn, and therefore the amount of time the first soldier is distracted, has been significantly reduced. The Green Beret dies in episode because he is caught by the first soldier, while carrying the body of the second. This playability error would appear as

an increased number of deaths in a traditional heat-map analysis, showing the undesirable result, but not the ultimate cause. In our PLATO analysis the diagramming serves to disambiguate the causes for game imbalance.

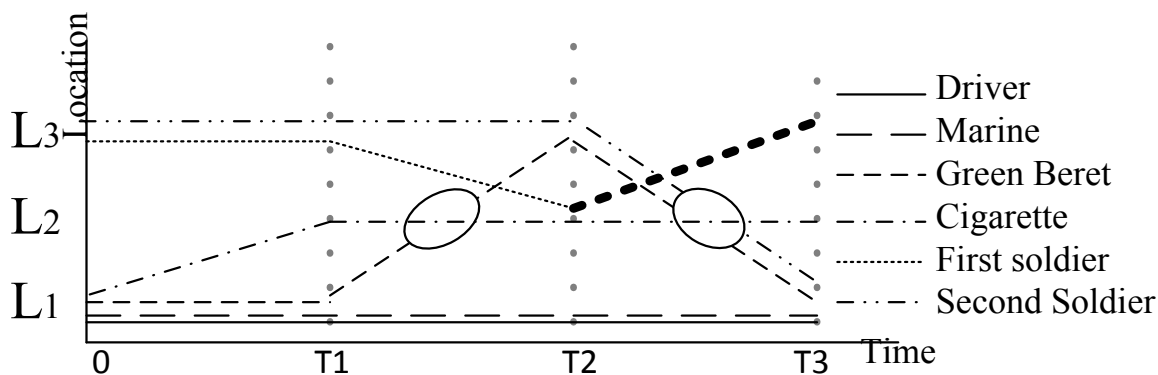


Figure 4-6. Timing error collaboration design. The thick dotted line represents the return of the first soldier which crosses over the dashed line indicating Green Beret movements.

In the LPT coordination diagram shown in Figure 4-6, the return of soldier 1 to location 3 before the Marine leaves is apparent, as is importance of the active coordination object (the cigarette). This error that is shown as a thick dotted line, illustrating the physical contact between the Green Beret and the first soldier. This issue can be solved by redesigning the Location-Time dependency of these game entities between T2 and T3.

Our second episode is imagined to take place during the design or paper-prototyping stage. In this case, the Marine has the capacity to both harpoon the guards and carry them away. In this case, the scenario collapses because the Green Beret no longer has a role. The Marine can simply kill both guards and drag them away with the help of the cigarette. If the level was played by two Marines, little coordination would be required.



Figure 4-7. The Commandos episode when Marine has the ability to both harpoon and carry dead bodies.

The collaborative paucity of this scenario stands out clearly in the Player-Time coordination diagram (Figure 4-7). Once the cigarette is thrown, the Marine kills and drags away both guards. Sketching diagrams like this one at an early stage can help design teams decipher if their collaborative scenarios have the desired degree of interdependency before the digital version of the game is built. In the next chapter, I will continue the game analysis process that takes place after the beta-testing and at the very last design iterations in a case study of the game Portal 2 co-op mode.

CHAPTER 5

COORDINATION IN PORTAL2 CO-OP MODE

Although multiplayer games exist across genres, the unique characteristics of Portal 2 co-op mode, particularly the deliberate and explicit pacing of collaborative acts, makes it an excellent choice to illustrate PLATO analysis. The co-op mode of Portal 2 is a highly collaborative puzzle game and has been designed in a way to make players to coordinate in the whole game scenarios. Coordination and collaboration is the only way for succeeding the objectives and solving the puzzles.

Portal 2 is a first-person puzzle-platform, developed and published by Valve Corporation. The game's robotic characters, Atlas and P-Body, are equipped with weapons which create portals – wormhole-like passageways which link two spaces – to solve challenges. In Portal 2, co-op mode consists of 5 chapters featuring individual test chambers which present puzzles that are designed to be impossible to complete individually. The objective of every test chamber presented here is to reach the exit without being destroyed.

5.1 Experimental Setup

Two players played the co-op mode for approximately 8 hours to train with the game mechanics prior to analysis. Both players returned to the beginning of the co-op campaign to replay the levels. As the puzzles had been solved once in the first run, the second run focused on using PLATO to analyze the coordination tasks. During the second run 20 scenarios were completed by players playing on computers in separate rooms, while screen capture recorded their progress. Three illustrative scenarios are selected and described here in detail.

Each chamber was analyzed using video annotation in two phases. In the first phase all collaborative elements in the scenario were identified and given labels. In the second pass, the timing and collaborative behaviors of the collaborative tasks were charted. As PLATO focuses solely on what players did rather than why they did it, no speak-aloud, retrospective or otherwise was required.

5.2 Interaction Tools in Portal 2

Multiplayer games are usually equipped with technical support for interactions among the players. In Portal 2 both voice and text chat options are available, but insufficient to communicate directionality in the virtual environment, key to solving the game's puzzles. Portal 2 takes advantage of three additional collaborative tools: a geotagging device which enables players to visually tag locations in the environment to inform their colleague where to create portals or where to move; a synchronized count-down-timer to support simultaneous actions, and the ability for either player to look through the eyes of their partner.

5.3 Game Elements

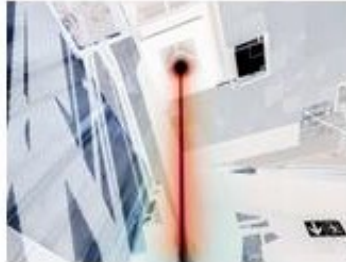
The Portal series is renowned for its innovative use of physics to create engaging puzzles, many of which involve manipulating objects. Weighted Storage Cubes and Edgeless Safety Cubes are passive objects which are typically used to hold down switches with a matching shape. Thermal Discouragement Beams are essentially laser beams which can be manipulated using Redirection Cubes. Turrets are simple stationary robotic objects that shoot at characters in their line of sight. Several other novel objects exist within the Portal universe but are not present in the scenarios described here.



Weighted Storage Cube



Edgeless Safety Cube



Thermal Discouragement Beam



Redirection Cube



Turret

Figure 5-1. Game elements¹¹ of Portal 2

5.4 Results

In this section I present three distinct scenarios addressed with the pattern of (Chapter, Level, Chamber). Coordination elements are charted to demonstrate the scope and utility of the PLATO framework.

¹¹ Source: <http://theportalwiki.com>

5.4.1 Portal2 - Scenario 1 (1, 1, 1)

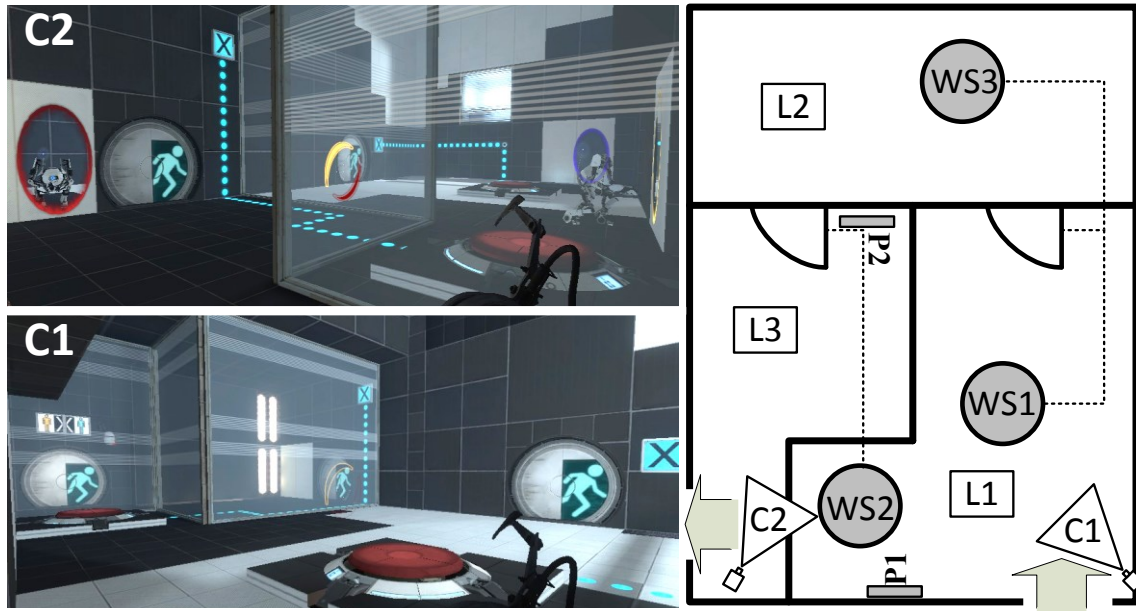


Figure 5-2. Game environment of the scenario (1, 1, 1). Cameras (C1 and C2) demonstrate two views of the environment from different positions. Three distinct locations are specified as L1, L2, L3 as well as the three weighted switches WS1, WS2, WS3. In addition, P1 and P2 illustrate the edges of a portal.

Table 5-1. Coordination elements for scenario (1, 1, 1)

Elements	Entities in the game
Players	P-Body, Atlas
Locations	First, Second and Third Rooms
Actions	Creating Portals
Time	Order ($T_7 - T_9$), Individual ($0 - T_2$), Bound ($T_2 - T_3$, $T_5 - T_6$)
Objects	N/A

Players start the scenario (1, 1, 1) (Figure 5-2, Table 5-1) from the first room of a series of three connected partitions and must proceed to the exit in the third room. The full analysis encapsulates two sub-scenarios.

The first sub-scenario involves entering the second room from the first. The parameter of general goal is entering to the second room and the specific goal is keeping the door, connecting the partitions, open. At (0 - T2): Player A should place a portal at a valid location (P1) in the first room (L1) (**PAL**). At (T1 - T4) Player B must stand on the weighted switch in the first room (WS1) which opens the first door (**PAL**). At (T2 - T3) Player A enters to the second room (L2) (**PL**). At (T4 – T7) Player B stands on the second weighted switch (WS2) in the first room which opens the second door (**PAL**). At (T5 – T6) Player A must move to the third room (L3) (**PL**). At (T7 – T8) Player A should place their second portal in the third room (P2) (**PAL**). (T8 – T9): Player B should use the portals (P1→P2) to move to the third room (**PL**).

Activity visualization of the PLATO diagram notation provides better understanding and explanation of the scenario's player coordination for designers. It provides an effective facility to explore, modify or design coordination in the unambiguous visualized domain. In this episode, initially activities can be assigned to either player but once a player has started the chain, they become Player A. Time and Location diagrams are shown in Figure 5-3 and Figure 5-4. This episode is a repetitive version of the hallway example described in section 4.1. The PAT diagram (Figure 5-3) demonstrates the dependency of actions on time and illustrates the underlying timing types. Figure 5-4 explains how the latter concepts are dependent to spatial attributes in a LPT diagram. Time (Figure 5-3) and Location (Figure 5-4) are the dependency factors of this example which implies that the activities are connected and interdependent via the factors of “T” and “L” (TL).

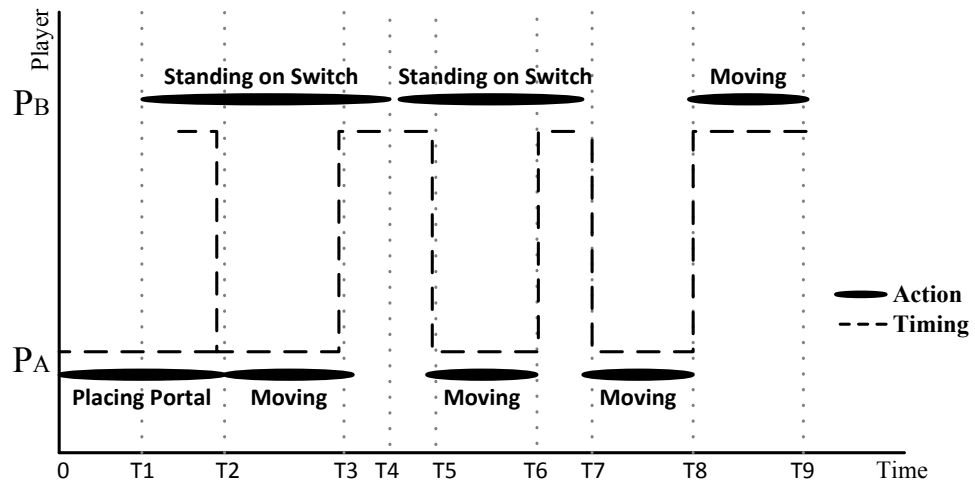


Figure 5-3. Scenario (1, 1, 1) - PAT diagram

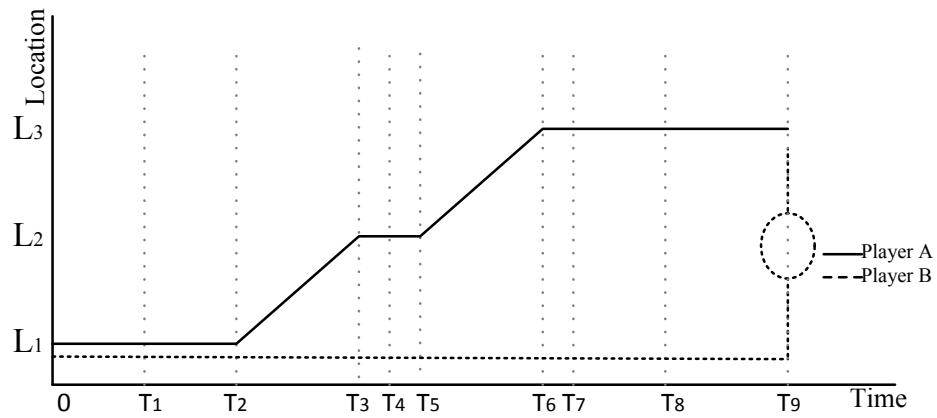


Figure 5-4. Scenario (1, 1, 1) – LPT diagram

5.4.2 Portal2 - Scenario 2 (1,2,2)

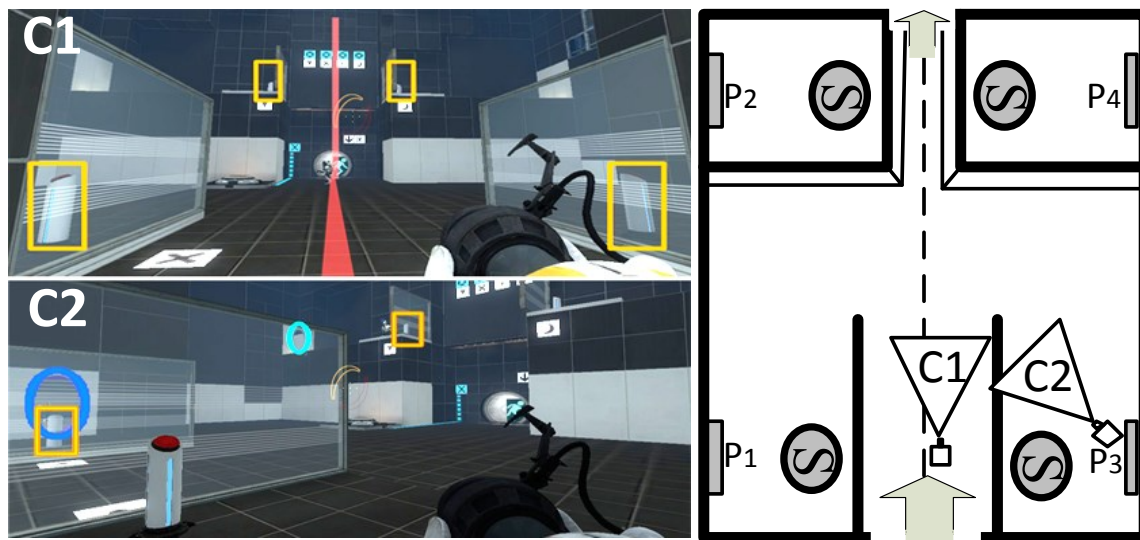


Figure 5-5. Game environment of the scenario (1, 2, 2). Cameras (C1 and C2) demonstrate two views of the environment from different positions. Four switches are specified as S. P1 and P2, and P3 and P4, each illustrate a portal pair.

In (1, 2, 2), players must perform similar tasks using soft-simultaneous actions by pushing a series of buttons in rapid sequence to release an Edgeless Cube. The room (shown in Figure 5-5) is completely symmetric with two buttons (S) on each side but at different heights. Each player must press two buttons in a short amount of time, requiring portals to cut the distance between two buttons; this implies order in soft-simultaneity. The general goal of this episode is releasing a cube and the specific goal is pushing the buttons simultaneously.

Table 5-2 illustrates the component analysis of this scenario. At $(0 - T_1)$ both players create a portal set which connects their individual button locations (P_1, P_2 and P_3, P_4) (**PLA**). At $(T_1 - T_2)$ Players push their first buttons (**PLA**). At $(T_2 - T_3)$ players push their second buttons (**PLA**). Finally, either of the players can unlock the final door.

Table 5-2. Coordination elements for scenario (1, 1, 1)	
Elements	Entities in the game
Players	P-Body, Atlas
Locations	Location of Buttons
Actions	Creating Portals, Pushing Buttons
Time	Order (T_1 , T_2), Soft-Simultaneity, Order in Soft-Simultaneity
Objects	N/A

Time is the only dependency factor of this episode. The timing diagram in Figure 5-6 demonstrates the independent action allowed prior to the first button press, then the rapid simultaneous action required for the second press. The final step of unlocking the door is not shown as it is not collaborative.

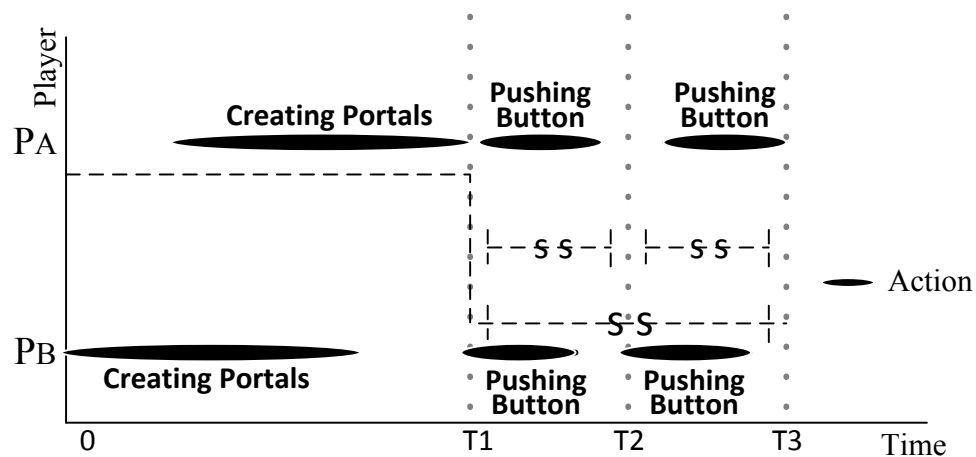


Figure 5-6. Scenario (1, 2, 2) - PAT diagram

5.4.3 Portal2 - Scenario 3 (1,3,2)

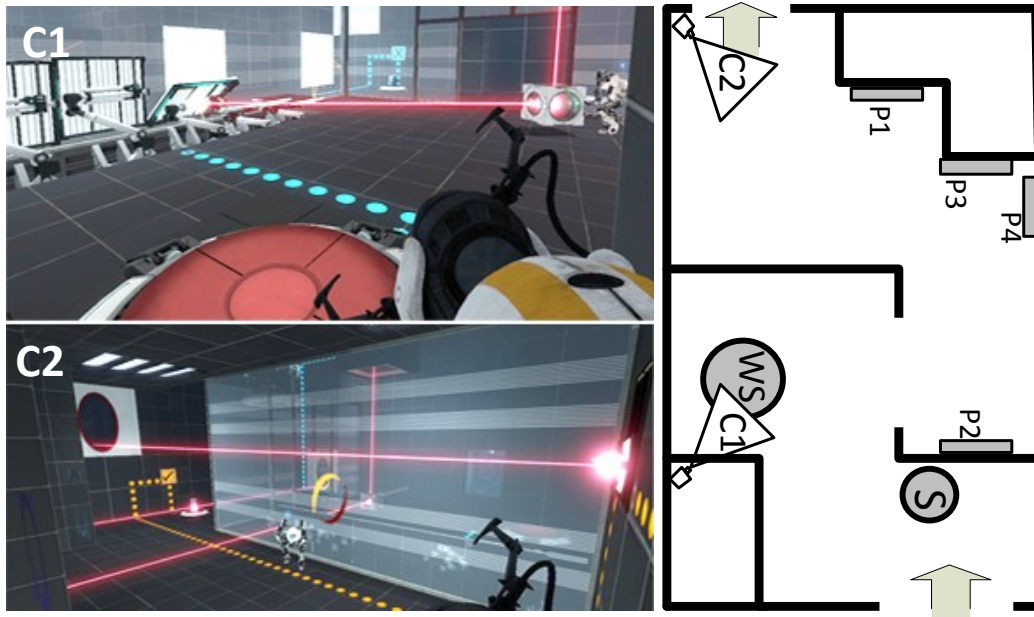


Figure 5-7. Game environment of the scenario (1, 3, 2). Cameras (C1 and C2) shows two renders of the environment from different positions. A switch (S) and a weighted switch (WS1) are indicated in the figure. P1 and P2, and P3 and P4, each illustrate a portal pair.

In this seemingly complex episode (Figure 5-7), the players must collaborate to repurpose lasers to both destroy turrets and unlock the final door; which comprises the two sub-scenarios in the test chamber. The analysis of scenario (1, 3, 2) is shown in Table 5-3.

In the first sub-scenario, players must obtain a redirection cube to aim an environmental laser at a series of turrets without being destroyed by either the laser or the turrets. The general goal of this sub-scenario is destroying the turrets and the specific goal is directing the thermal discouragement beam toward the turrets. At (0 - T_1) player A should push a button (S) to release a Redirection Cube (**PLA**). At (T_1 - T_2) player A should move to the location of the cube (**PL**).

At ($T_2 - T_4$) player A places the cube under the thermal discouragement beam (**PALO**). At ($T_4 - T_5$) player A should change the direction of the cube to aim the beam towards the turrets (**PALO**). Over ($T_3 - T_6$) player B should stand on a weighed switch (WS) allow player A to see the turrets (**PAL**). Dependency factors of this episode are Time and Object.

Table 5-3. PLATO elements for scenario (1, 3, 2)

Elements	Entities in the game
Players	P-Body, Atlas
Locations	Laser sensitive area, accessible part of laser beam, location of the cube, the area of the exit door, location of portals, location of the button
Actions	Creating Portal, Changing the cube's position and direction, pushing button
Time	Individual ($T_0 - T_4$), Bound ($T_4 - T_5$), Parallel ($T_6 - T_7$), Order ($T_6 - T_8$)
Objects	Redirection Cube

The only collaborative act in this sub-scenario is the division of labor, where one player must open the window to expose the turrets while the other destroys them with the laser, illustrating the Bound constraint. Player B must stand on the switch before Player A starts destroying turrets, and must leave the switch after the turrets are destroyed, but is otherwise unconstrained in time.

The second episode involves directing the laser through a series of portals to a laser-powered lock opening the final door. The general goal of this sub-scenario is reaching the exit door and the specific goal is leading the laser beam to a specific location. At ($T_6 - T_7$) players should place two portal sets (P_1, P_2 and P_3, P_4) (**PAL**). At ($T_7 - T_8$) either player should aim the laser beam through the portals toward the lock using the cube (**PALO**). The primary collaboration in this instance is coordinating the placement of the portals, which is time independent and tied to relative position. What appears to be a complex episode, involving

multiple objects, switches and actions, devolves to a relatively simple coordination episode in the diagram shown in Figure 5-8. Player B is irrelevant until T_3 , then must stand on a switch while Player A completes the action. The subsequent parallel portal creation must be performed correctly but is limited in temporal complexity. Location and Time are the dependency factors of this scenario.

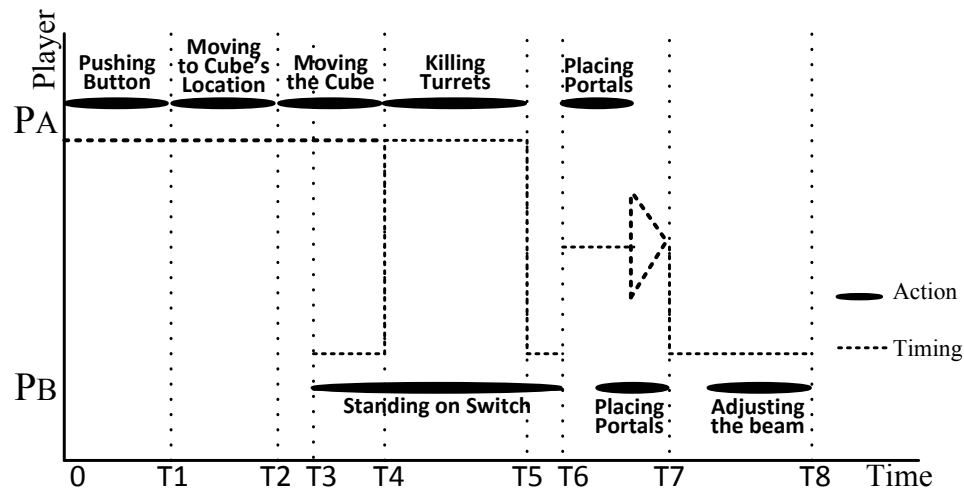


Figure 5-8. Scenario (1, 3, 2) - PAT diagram

CHAPTER 6

PLATO AS A COORDINATION DESIGN TOOL

In the previous chapter I showed the strength of PLATO in analyzing collaborative episodes and explained how this structural analysis can support designers in evaluating and adjusting coordination before playtesting. I performed the analysis first by decomposing a game episode to atomic actions and thereby extracting underlying coordination elements in each atomic action. In this chapter I continue the investigation in collaboration and player coordination by focusing on the design process to show how the framework can assist designers in the process of coordination design.

Although digital games often are individual with limited collective action [70], multiplayer games leverage players' collaboration and interaction. In a game scenario with interdependent activities, players undertake different divisions of tasks with respect to others' actions based on the designed coordination [19].

In contrast with the analysis process explored in the last chapters, the act of coordination design in PLATO follows a recursive procedure starting with more general definitions of the scenario and leading to more specific definitions with additional detail. The general goal is the starting point for designing a complete collaborative scenario, specifically using the PLATO diagrams and tools, coordination and dependency factors can be defined and visualized at any stage of the design process.

In this chapter I investigate the process of design through several different design and analytic scenarios for a real game example. I demonstrate the strength of the PAT diagram in the first stages of coordination design by providing a high level abstraction for a clear illustration of

coordination. Using the LPT diagram I show the process of mapping the designed coordination from the PAT diagram to spatial attributes. I design a collaborative game prototype based on a set of social design patterns and in the next step, evaluate the prototype in terms of player coordination using the PLATO framework. In the second approach I continue the process of design and use PLATO to analyze the results of the first prototype.

Kreimeier [36] for the first time raised the need of a shared vocabulary and formal method in designing games. By taking advantage of Alexander's idea [1] of formulating frequent problems and solutions as design patterns, he presented a set of game design patterns. Bjork and Holopainen [4] also presented a set of design patterns with the focus on gameplay to facilitate game evaluation and design processes:

“Having a language of gameplay that can be used for game design is not the total solution for designing good games... it does give you a tool to help your thinking and communication; a tool that can help you put words to an idea or concept and either write it down for future [4, p4].”

A multiplayer game may not be a pure collaborative game. A team of players might collaborate while competing with opposed teams, in other words, “Mutual assistance is usually the only way that the players can reach the objective... Most cooperative games are competitive in nature but the rules or goals of the game force the player if he wants to win” [70, p5].

From one approach, a design of a multiplayer game should include spontaneous and stimulated social interaction as the extent the game environment facilitates players' interaction inside and outside the game or stimulates it by rules and design features, cooperation and competition, synchronicity or concurrent actions, coordination, dependency and meta gaming caused by information asymmetry [70]. Designs with good visualization to support presence by symbolic representation of players can effectively support spontaneous social interaction and

self-organization [65]. Based on the playground game Cops and Robbers, I designed a game in collaboration with A. Tavassolian. To leverage team play we assumed a team of three cops cooperating and collaborating with each other and competing with a single hacker. They pursue incompatible goals while cops with symmetric goals try to traverse the playground and save the computers from being hacked, the hacker tries to capture the computers and evade the cops. Seven locations of computer systems were defined in the game and the hacker pursues the goal of hacking systems one by one and to win the race. To support players' awareness where face-to-face interaction is not possible, the cop's interface is designed to provide information such as embodiment and presence of the other players. The cops are also equipped with a chat interface with the ability of overhearing other cops' communications to leverage players' interaction. The game is balanced such that catching the hacker is unlikely with a single cop. Instead, cops should approach physically to the location of the current attacked system (the current location of the hacker) and collaborate to arrest the hacker.

6.1 Stealth Hacker

Stealth Hacker is a mixed-reality location-based game inspired by the playground game Cops and Robbers. The game is played with several cops and a single hacker. The game's playground is a shared network of computers. The cops investigate the playground physically, visit computers and search the systems with their smartphones (Figure 6-1) while the hacker is trying to infiltrate the network. The hacker navigates the playground virtually by moving his simple avatar in the network diagram and changes his place from computer to computer (Figure 6-1).

The cops' interface supports their awareness by providing them with information on the location of other cops (current position and planned movements), a chat interface and the hacker's last known locations. Cops move from one computer to another and scan them using a smartphone. The smartphone does not actually scan a computer rather it records the computer's Bluetooth MAC address, and send it to the server wirelessly. The server then updates the game state for that computer to "scanned". The process of scanning and hacking are simulated as minigames on the hacker's PC or smartphones. The Hacker's objective is to hack all the computers in the network. The cops' objective is to catch the hacker. To catch the hacker, cops need to be at the same physical location as the hacker's virtual position and 'scans' all the computer before the hacker can escape.



Figure 6-1. In this figure, on the right, cop's interface on a smartphone and on the left, hacker's interface on pc is illustrated.

6.1.1 Stealth Hacker's gameplay evaluation

Four participants played the game for nine rounds. MacBook computers with specified Bluetooth MAC addresses were utilized as the physical nodes. These nodes were situated over

two floors in the University of Saskatchewan Computer Science Department. Balancing the game timing between the player in the virtual world and real world was one of the major design processes. The game was equipped with a number of minigames launched during the game for the hacker when the hacker begins hacking a system or was scanned by a cop. These minigames with adaptive timing feature manage the game balance in the mixed game world. According to the PLATO framework, the coordination elements involved in the game are Player, Location, Action and Time. The activities of players in this game are explained as:

PL: When cops change their locations from one computer to another computer

PLA: When cops trace the location of the hacker at a certain system.

PLAT: When cops trace the location of the hacker at a certain system together soft-simultaneously to capture the hacker.

Figure 6-2 shows the LPT diagram of the three cops in a single run of the game which illustrates their team effort in chasing and catching the hacker in the playground consisting of seven distinct locations. The first three nodes were locating in one lab and the last three nodes in another lab while the node number 4 was located in between. According to the diagram, the cops started the game in different locations and I can see them converging to a certain location in the game trying to catch to hacker. At ($T = 329$) cops caught the hacker using teamwork.

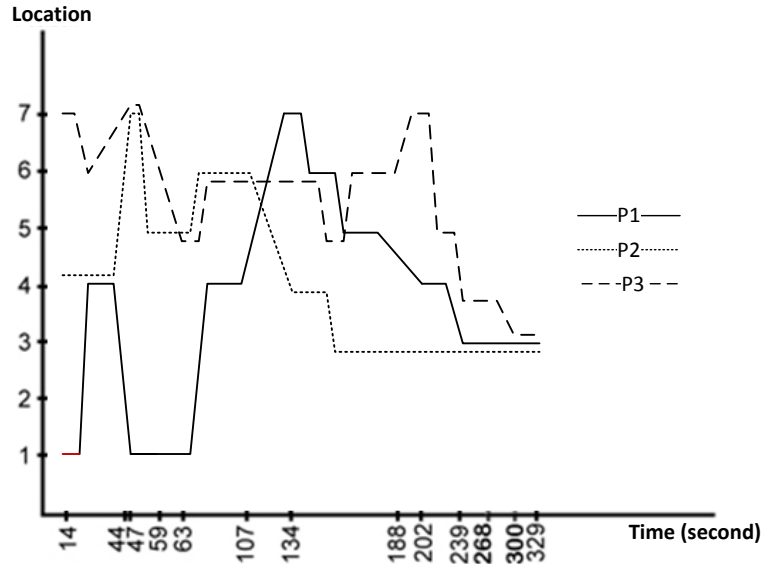


Figure 6-2. LPT diagram in a single run of the Stealth Hacker game. Each player (p1, p2 ,p3) are specified with a distinct line pattern. There are seven nodes in the playground at different locations.

At ($47 < T < 59$) while the hacker was hacking the system at L7, P2 quickly reaches P3 to coordinate in catching the hacker. At ($T=107$) P1, after chasing the hacker, joins the physical locations of the two other players. Finally they converge at (L3, $T=300$) to collaborate in tracking and arresting the hacker.

6.2 Stealth Hacker V2.0 (SH2)

In Stealth Hacker, I aimed to design a collaborative game where players are able to coordinate their actions and collaborate to achieve the game objective. To support player coordination and social interaction, as described in the previous section, the game was designed based on a series of known and common patterns used in designing co-op multiplayer scenarios. Though design patterns support the process of design effectively, they are usually an abstraction

which does not contribute explicitly to the process [4]. In the next section I use the PLATO framework to analyze and adjust the game which leads to redesign and create a newer version of the game.

6.3 Stealth Hacker Analysis

Social interaction in games is supported and affected by a variety of different factors specific for each game. There is often not a single solution when playing a well-designed game and it provides different experience to the players each time they play the game [50]. The collaboration and communication within the game take place often at random moments in the game. Our process of design and manipulation of coordination in one approach can affect the overall gameplay or it can be as specific as defining certain collaborative activities at particular milestones in the game. This implies LPT or PAT diagrams usually do not describe the exact time that activities and events take place in the game. Instead they provide a general overview of the order and specifications of the coordination and gameplay. In the process of designing an episode, the diagrams should cover different possibilities of activities in that episode, and the general trend and order the activities will be performed by the players. PLATO tools provide rendering of the episode including the coordination information underlying the scenario to be used as a coordination design tool without limiting the gameplay to a particular form.

The Stealth Hacker game can be explored in different perspectives. Comparing the experimental results with the design specifications and our expectations of the game can clarify the successful design in supporting player coordination and interaction. Based on the results and observations, Stealth Hacker was able to engage participants within a cooperative game. In some aspects, the results met the expected criteria and for some others, it did not satisfy the

coordination design goals. In the next section, I explore some of these design features by evaluating the gameplay and the impact of dependency factors on the gameplay. Using PLATO, I illustrate how coordination issues can be addressed by redesigning coordination first by manipulating the general gameplay, and second, by focusing on coordination at specific milestones.

6.3.1 Coordination design by general gameplay manipulation

6.3.1.1 Location dependency

Location is a dependency factor in Stealth Hacker which implies that cops should appear at certain locations in the game environment in a coordinated fashion. As the hacker is able to traverse the game world quickly by using the keyboard and moving from one system to another, I considered a larger scale playground that forces the cops to distribute physically in the given space and interact without using face-to-face communication. However, as shown in Figure 6-2, although they started the game maximally distant from each other, they converged physically in the game. To make it simpler, I assume the case with three locations (the two labs and the area in between). I start the coordination design process first by defining the general goal as a territory game by assigning a territory to each cop. To visualize this scenario and be able to explain our design expectation of the way players move in the playground I used the LPT diagram (Figure 6-3).

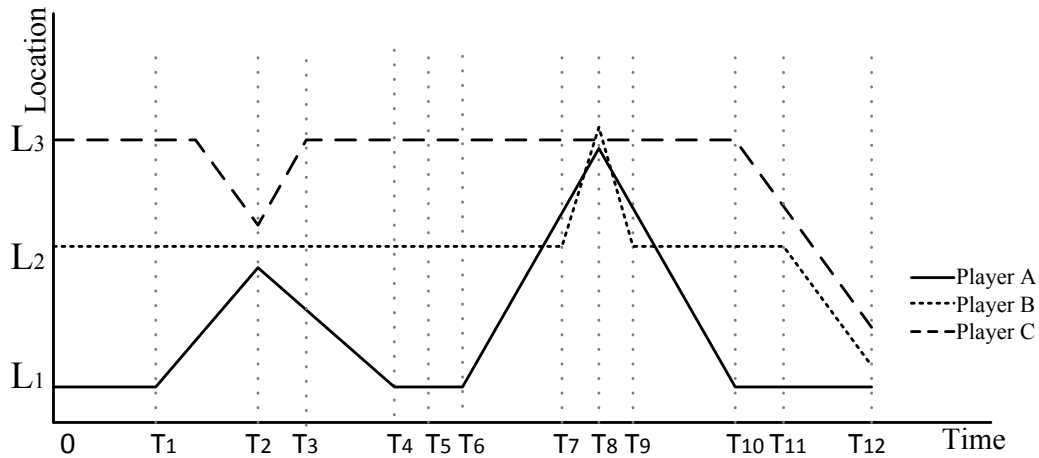


Figure 6-3. LPT diagram of an hypothetical desired game trend.

Figure 6-3 represents a LPT diagram of a hypothetical game. The diagram shows the dependency of players to specific locations and physical convergence resulting in the hacker's capture. In this hypothetical diagram, players find a clue about the hacker's position in L_2 , leave their current locations and move to L_4 at T_2 . After an unsuccessful chase, they return to the assigned locations and continue to coordinate with each other. Another unsuccessful chase is considered at (L_3, T_8) and finally players catch the hacker at (L_1, T_{12}) .

A comparison between Figure 6-2 and Figure 6-3 shows that although players have location dependencies in Stealth Hacker, this dependency refers to any location and not a specific location. In designing SH2 gameplay I will use information asymmetry, which implies that players (cops) will have limited information when chasing the hacker. In other words, I define the specific goal as information gathering. The game map will not inform them of events in the whole game world; instead players will only be informed of events at nearby locations. For example, the player at L_1 will be informed of the hacker's activities on the three computers located in that area. Cops should need to traverse and cover a large fraction of the game world to

win the game, forcing them to maintain their distance to cover the field. To further complicate coordination, players could be assigned a specific location at the start of the game and either keep this attachment during the game (Figure 6-3) or change it dynamically. In PLATO, this takes place by adding Player to the set of dependency factors because a specific player aspect (territory assignment) affects game state and coordination strategy. The dependency factors will be (PL) in which P and L are connected to each other. Players should carry different roles or abilities which are location-dependent. For this specific scenario, this design goal can be achieved by assigning location-dependent asymmetric information to the players. In this case, each player is in charge of a specific area and needs to guard their own area.

6.3.1.2 Increasing the interdependencies

Another approach to coordination design is increasing coordination by adding more interdependencies. By analyzing Stealth Hacker, it can be said that the game is more cooperative than collaborative in some aspects: although players coordinate to catch the hacker, players spend the majority of the time searching independently (PL). Injecting another dependency factor in the gameplay will make the game more collaborative and change the dependency factors to PLO. The prior analysis of Stealth Hacker revealed three distinct atomic coordination actions in the game (PL, PLA and PLAT).

In Stealth Hacker, each cop was able to track the hacker independently using their own device. I make a slight change in the game by making the tracking device a shared resource. In this case there is a single “tracking device” that must be physically passed between the cops to track the hacker locally. This is achieved by only changing the specific goal describing how to achieve the general goal. The general goal is still “tracking the hacker” and the specific goal

explains that using the new tracker object will be the only way to track the hacker. Any cop will be able to carry and utilize the object and it can be passed from one cop to another. Other cops will be still carrying the same physical device they used to carry but without the ability to track the hacker's location (only trap the hacker when found). This change implies two different role-specific action types. The two cops who are carrying the smartphones with limited location-specific information explore the game world and the one with the tracker tool, using the information and clues given by the two other cops, should follow and trace the hacker's location. Figure 6-4 shows the PAT diagram of a potential game after applying the shared resource change.

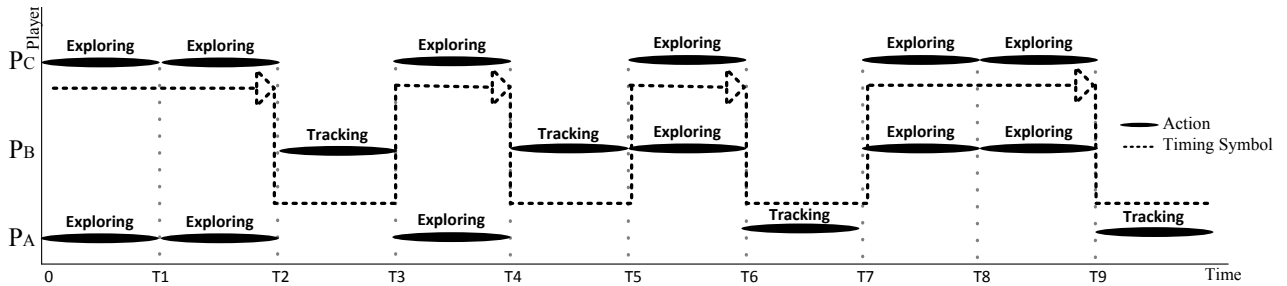


Figure 6-4. Stealth Hacker PAT diagram of an expected game current by using a shared resource

According to Figure 6-4, the timing of the game scenario is now hybrid parallel-order time. In this diagram, players start the game at $T=0$ while P_A and P_C carry the cellphones and P_B is equipped with the tracker object. At T_2 , P_B receives clues about the hacker's position from the two other cops and starts the activity of tracking the hacker's location. This helping behavior [15] results in the illustrated increase in stimulated social interactions and in [56] is referred as a cooperative performance metric.

The unsuccessful tracking operation at T_3 reveals the hacker has left that location which makes the cops to continue to chase the hacker. At (T_4-T_5) the tracking action takes place again. As tracking needs to move rapidly from one place to another, I expect players to change their roles frequently. At T_5 , P_B and P_A exchange their roles and the tracking action is performed by P_A .

6.3.2 Coordination design at certain milestones

The process of design and manipulation of player coordination and collaborative activities can also affect the game at specific milestones without a significant change in the game. This approach support the design process by managing coordination scope within the scenarios such as increasing player coordination and interdependent actions within a scenario which suffers from insufficient social interactions and collaborative activities or in contrast, reducing the complexity of highly interdependent scenarios by redesigning and correcting the underlying coordination. It also can help designers in designing a series of collaborative episodes to be used in the process of game design.

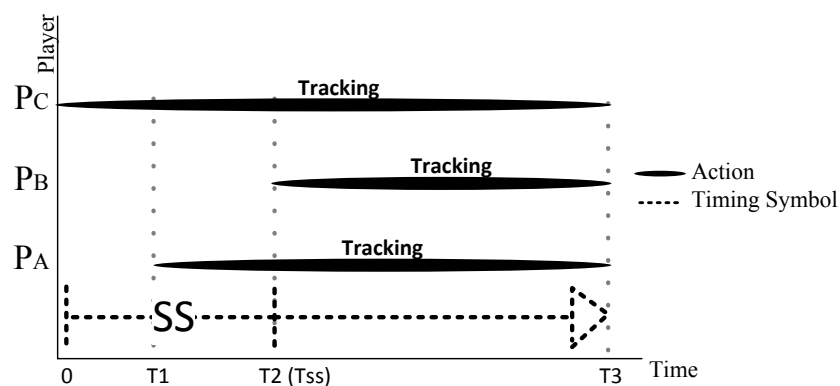


Figure 6-5. Tracking hacker's location

In this section, I continue the design process of SH2 by examining the game play at specific milestones. As an example, the process of tracking the hacker's location can be redesigned to adjust the underlying coordination. Figure 6-5 shows the analysis of the current tracking episode within a PAT diagram as an ideal tracking process in which all the three cops meet the hacker's location in a soft-simultaneous time window. At $T=0$, P_C initiates the tracking process and at $T=1$ and $T=2$, P_A and P_B join P_A within a soft-simultaneous timed collaboration. As the figure illustrates, from T_2 to T_3 the tracking process continues as a simple parallel timed activity.

6.3.2.1 A simple coordination design

Redesigning this episode using PLATO would be a flexible and straightforward process by only altering the specific goal and without making any change in the current dependency factors and the general goal as "tracking the hacker". Time as an essential aspect of coordination which shapes the collaborative activity plays an important role in defining the specific goal. Applying a change to the time factor in an episode results in changing the way people coordinate within the current coordination elements. For example, for this specific scenario, the coordination elements are described as PLAT in which some players converge physically at a certain location and within a soft-simultaneous task perform the action of tracking. Manipulating the timing factor of this scenario will lead to changes in the way collaboration unfolds while other elements such as location and player are not affected. By switching the time elements with another timing as shown in Figure 6-6 I redesigned the episode to encapsulate a different hypothetical interaction.

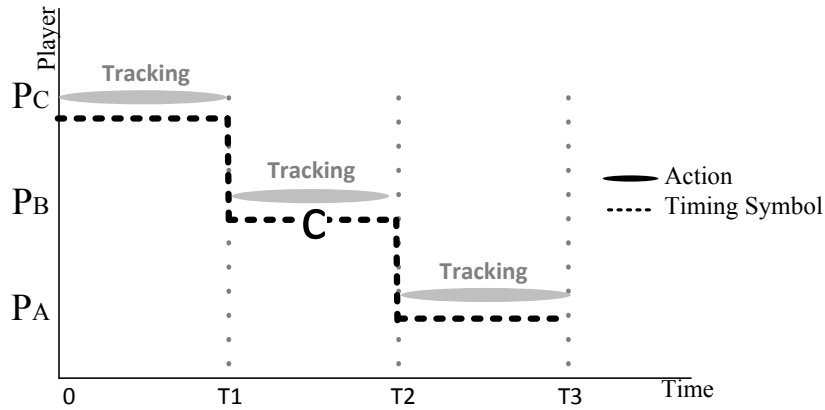


Figure 6-6. An hypothetical simple coordination design

Figure 6-6 shows our first attempt in changing the coordination. In this episode players are not doing simultaneous actions and instead, they track the hacker in a certain order. For this new case, a single cop is not able to succeed in the task as the task takes longer and the process needs to “stay alive” by another cop when the previous one has done their task. That means in the new scenario, cops have limitations in tracking the hacker and are able to continue this process for a limited period each time which is not long enough for catching the hacker. As the time limitation of this new scenario is lower than the original scenario (comparing soft-simultaneous timing to clocked-order timing) cops have more time to reach the hacker’s location and join the tracking process. However, when a cop is done with their activity, the other should be at the location to start their activity and keep the tracking operation alive.

6.3.2.2 More complex coordination design

Designing player coordination in a collaborative episode can be adjusted from a simple straightforward process to a more complex collaboration by increasing the timing complexity.

Figure 6-7 shows another attempt at redesigning the episode with more complexity. In this episode, one player is supporting the two other players who are doing simultaneous tasks. This is achieved by a combination of bound and simultaneity timings according to Figure 6-7. To implement this episode in SH2, the defined actions need to be updated to match the game story and meet the requirements addressed in the PAT diagram. I assumed a conceptual tracking communication channel for tracking the hacker which needs to be kept open by a player while the two other players track the location. In this case, as Figure 6-7 illustrates, P_A and P_B perform the same task of tracking simultaneously, while P_C is supporting their synchronous actions by keeping the line open.

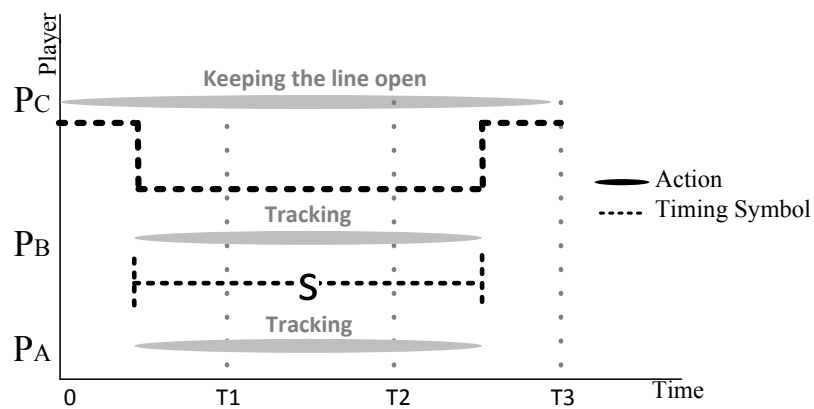


Figure 6-7. An hypothetical more complex coordination design

6.3.2.3 Another hypothetical design example

Similarly, this approach can also be the start point of the coordination design process. The advantage of this method can be described via the abstraction provided for designing player coordination independent of the other settings and elements. Figure 6-8 shows the third attempt to design this scenario. According to the time table (Table 4-1), from 0 to T2, PC is performing a

solo activity of individual order (two actions in a row). The two other players will coordinate with the first player and start their collaboration when the first player is done. The two other players start coordinating on a bound task in which one player will support the second player's activity. Defining the correspond activities and mapping them to the game current and story can be done subsequently.

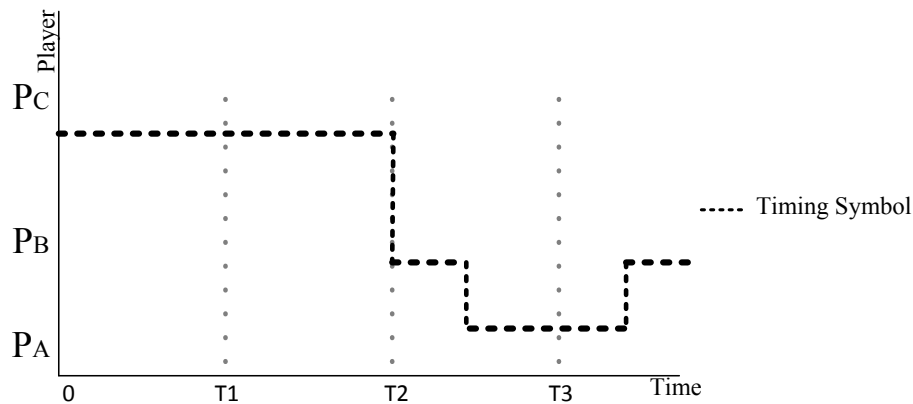


Figure 6-8. The third attempt in redesigning the scenario

6.3.2.4 The key role of PAT and LPT diagrams

I conclude this section by designing and adding one more collaborative activity to SH2. The main goal of this scenario is to demonstrate how the diagram notation facilitates the process of design. The level of abstraction provided by the PAT diagram enables us to define the collaborative activities by focusing on coordination. In section 6.3.1.2 I changed the tracking activity by adding a tracker object to the game. For this case, shown in Figure 6-2, two players often experience an idle time while the third player tracking. In this section I plan to address this idle time and coordinate the players within the ongoing tracking activity. To design this coordination I start from a PAT diagram shown in Figure 6-9 and begin the process using the time table (Table 4-1). To shape this diagram, two order timing symbols in a row and two

simultaneous activities are considered for two players using Table 4-1. Now it is possible to define the actions for the defined settings. To keep the design process simple, I defined the action of pushing a button for all the activities. As the action of pushing a button is a short-duration action, I defined a cycle in the PAT diagram and put it in the repeat box, shown by the letter “R”. That means this cycle repeats continuously while the process of tracking continues.

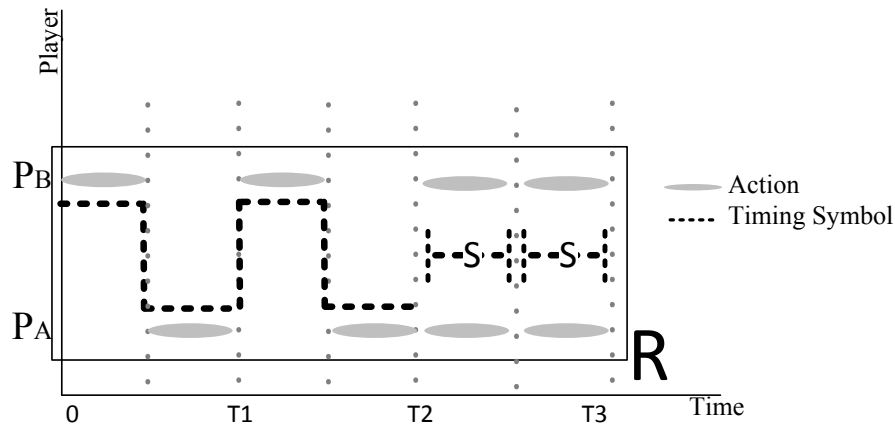


Figure 6-9. Coordination design for the two idle players shown in Figure 6-2.

Now it is possible to include the newly designed activity into the scenario and match it with the tracking process perform by the third player (Figure 6-10).

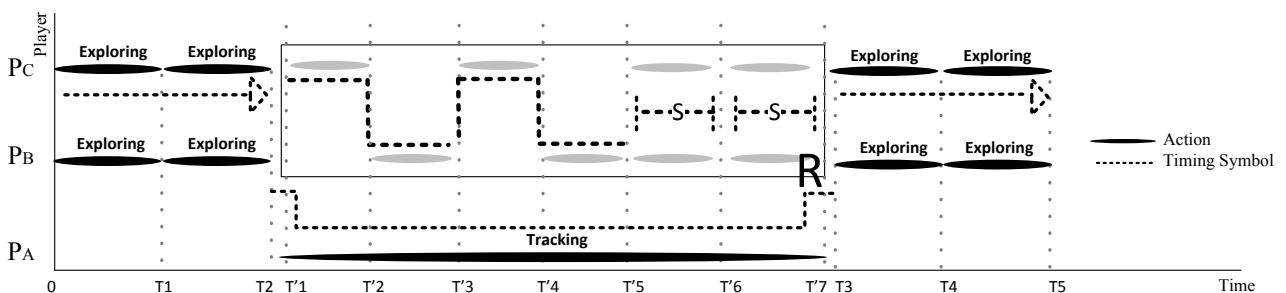


Figure 6-10. An overview of the three-player activities in this new scenario.

According to the new PAT diagram shown in Figure 6-10, in order to succeed the process of tracking, two other cops need to coordinate to provide support (bound timing) for the tracker cop by pushing the buttons alternatively as the PAT diagram shows. This can be matched with the game story similar to the concept of keeping the communication line open by the two cops.

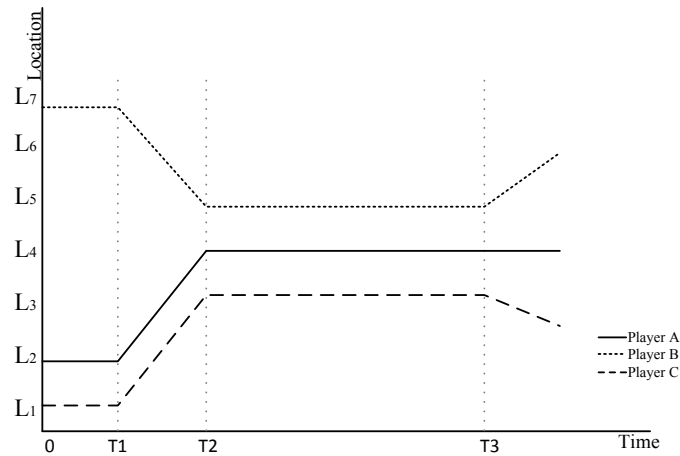


Figure 6-11. LPT diagram representing the spatial attributes of this scenario.

The designed coordination in the PAT diagram also needs a connection to the spatial attributes in the playground. Using a LPT diagram, I can define the spatial specifications. For this episode (Figure 6-11) the collaborative activity is tied to the adjacent computers of the system being hacked. Time as the common factor between these two domains facilitates the process of mapping and coinciding LPT and PAT diagrams. As the LPT illustrates, P_A moves to L4 from T1 to T2 to start the tracking operation. At the same time, P_B and P_C also join him at the adjacent locations and follow the operations from T2 to T3 according to the PAT diagram.

The examples presented in this chapter were brief explanations of the PLATO framework's contribution to the design process by creating an abstraction layer over the details and the complexities of game design. PLATO provides a conceptual game-specific coordination design environment for designers to concentrate on the concept of coordination design within a

clear and quantified procedure which was previously an abstract, try-and-error, and ad-hoc process.

CHAPTER 7

DISCUSSION AND DIRECTIONS FOR FUTURE STUDY

The examples and case study demonstrate the value of the PLATO framework for describing, analyzing, and designing coordination scenarios in multiplayer games. The portal case study demonstrated how PLATO can contribute to the last iterations of game design. I evaluated the strength of PLATO in designing player coordination in a real game scenario and the way it contributes to earlier design phases. It demonstrated different methods of coordination design by adding collaborative episodes at certain milestones in the game or by affecting the general gameplay and exploring different possible coordination factors with impacts on gameplay. PLATO illustrates different approaches for designing and analyzing player coordination. By comparing the design process with the coordination analysis methodologies, the act of coordination design in PLATO follows a recursive procedure which starts with the more general definition of the scenario and leads to more specific definitions and details in each step. The process of coordination design can be started by defining a clear general goal of the scenario. PLATO diagrams and tools can provide an opportunity to design player coordination and define dependencies at the very early stages without including the exact specification of the current game elements and specifications of player's actions.

The framework provides a consistent representation of one aspect of collaboration in games that has previously been difficult for designers to conceptualize, discuss, and evaluate. PLATO builds on previous frameworks by Malone and Crowston [40] and Eccles [19], but adapts the structure of the framework to reflect the differences between workplace and digital-game coordination. The perspective of the framework is the realization that whereas workplace

coordination is grounded in the requirement to complete a task (with measures of success related to the quality, speed, and efficiency of task completion), in games the process is the outcome, and therefore a more nuanced description of the elements of coordination is required. Temporal requirements in particular can have many different instantiations because of the real-time nature of the activities, and because the timing constraints in games can be used to change the experience of the scenario.

The design examples demonstrated the capacity of our framework to provide practical insight during game design and testing, by providing a common language, a diagramming nomenclature, and a structured analysis technique based on these artifacts. The goal of the PLATO framework is utility for designers, so the diagrams and descriptions built from the framework are descriptive. In addition, there is considerable flexibility in the framework for designers to contextualize the idea to fit the particular domain of their game. For example, there is space within the concept definitions for designers to alter the distinction between player and object, in a manner that is consistent with their game and genre. Given the complexity of digital games and the breadth of game types, there are necessarily many grey areas in any classification scheme. With PLATO I do not provide a rigid decomposition, but rather a tool for describing and analyzing multiplayer interaction, something that has been difficult for designers to do in the past.

Though the major focus of PLATO is on the design process of digital multiplayer games, this is not a constraint for the framework. We demonstrated this earlier by the investigation on Stealth Hacker as a mixed reality game in the previous chapter and illustrating the way PLATO contributes in the coordination design process of this mixed reality game. However, depending on the game specifications, using PLATO may apply a considerable overhead in the process of

design for some particular scenarios. A clear example of this situation is a simple card game in which players' interactions take place based on a series of play cards and the game follows a routine procedure including a set of rules, states and conditions. Due to the simplicity and the clear scenario of this game, using PLATO such as considering cards as objects and decks as locations will result in overhead in the design process.

Change in player coordination can alter game play complexity and affect game balance. In the section 4.3 we demonstrated how PLATO diagrams can be used to adjust game balance and be useful for revealing game play issues. As a descriptive solution, PLATO is only one of many possible techniques. I provide what I believe to be a compelling and useful general framework for the analysis of coordination in digital games. However, other general descriptions are possible, and extensions and refinements of PLATO could be useful for specific genres of games or classes of interaction. For example, different refinements of the framework could be developed for the more loosely-coupled coordination and asynchronously-evolving gameplay in FarmVille [21] or MafiaWars [38], and another for the tight real-time gameplay of an online shooter.

7.1 The unique approach of the PLATO diagram notation

Social interaction support mechanism often differs from one game to another and there is not certain pre-defined procedure to be used in the game scenarios. A well-designed game provides different experience for the players each time they play the game [50]. Depending on the way players interact with the game world and other players, interactions such as collaboration and communication take place at loosely specified moments in the game. This implies PAT and LPT do not describe the exact time that activities and events take place in the game, and the diagrams are not intended to limit players to a specific procedure to accomplish

the game objectives. Instead, they provide the general trend and order of the activities that should be assigned and performed by the players. PLATO tools provide rendering of scenarios which describe the way players collaborate in an existing game or in an under-development game. They provide a general overview of the order and specifications of the coordination and the gameplay. The process of design and manipulation of coordination in one approach can affect the overall gameplay or it can be as specific as defining certain collaborative activities at particular milestones in the game.

Among the current existed visualization tools and diagrams, Gantt and PERT are two popular charts that are frequently utilized in a wide range of commercial projects. Gantt charts for visualizing project schedules are bar charts which provide a clear overview of the estimated project trend [67]. PERT charts as project management tools are also used for task organization purposes. PERTs consist of the project milestones and demonstrate the dependencies of tasks on each other and the resources. The horizontal axis of the Gantt chart usually is time and each task in the chart is specified by the associated time. The type of timing attribute of the chart demonstrates the order and concurrency of the tasks and the vertical is assigned to the given tasks. Task dependencies such as critical paths and slack regions can also be shown in this chart.

Similar to a Gantt chart, the horizontal axis of the PAT diagram in PLATO is assigned to time. However, the time unit in a PAT diagram usually refers to short durations, often seconds to match the requirements of the real-time games. Due to the specifications of the PAT diagram which is designed for diagramming the players' coordination in the real-time games, it follows a different approach than the Gantt and PERT charts. PAT diagrams focus on the way players coordinate which makes the actor element an important factor. To address this key point, the vertical axis of this diagram is dedicated to the element of Players as the actors of the

collaborative scenario, and the action factor is included in the other dimension of the diagram. This approach enables the diagram to visualize the dependency of the actions not only to time but to the coordinating players. It enables designers to focus more on players' interaction by defining actions based on time. Due to concentration of real-world projects on productivity, Gantt and PERT charts illustrate activities based on time which makes an outstanding difference in the approaches of these diagrams.

Although Gantt and PERT charts are originally designed for project management purposes, Figure 7-1 illustrates an approach of using these charts to analyze the hallway example explained in section 4.1. As Gantt and PERT are activity based charts and do not provide spatial information about the scenario, they are not comparable with LPT charts. The PERT network shown in Figure 7-1 demonstrates the scenario as a graph of connected states. The dotted arrows highlight the need for returning the initiative state of the current activity. As Figure 7-1 illustrates, in the T_1 time window for the Gantt chart, the first activity starts and triggers the second activity. At the end of the time window, termination of the second activity is required to finalize the first activity (the dotted arrow) which, like the PERT network, is not supported in this type of chart. At the start of the T_2 time window and by termination of the second activity, the third activity starts and triggers the fourth activity. Due to the focus of Gantt and PERT charts on only activity and time, there is no information about the actors of the scenario and the chart does not visualize the relations and interdependencies of actions based on their actors. This is in contrast with PAT diagrams which focus on a variety of different time-dependency specifications. Gantt and PERT charts timing information is limited to general dependency types of order and concurrency.

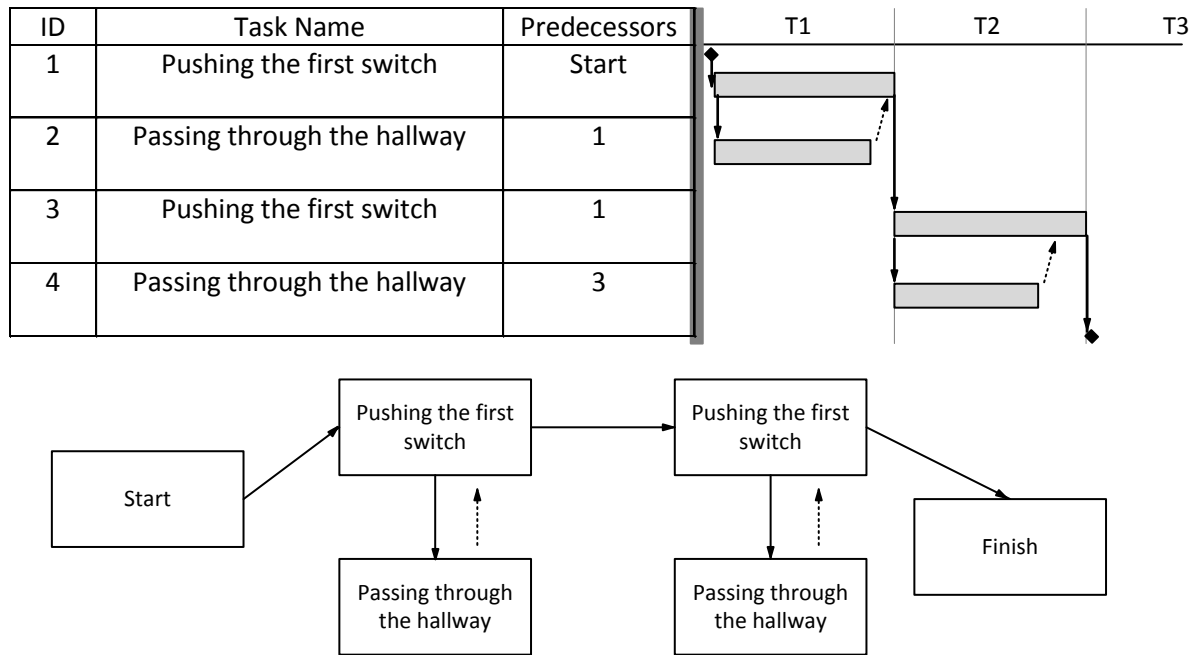


Figure 7-1. Gantt (top) and PERT network (bottom) charts of the hallway example in section 4.1. The dotted arrow in the Gantt chart shows the dependency of an action to its initiative activity.

7.2 Specific features of PLATO diagram notation tools

Apart from the different approaches of PLATO coordination diagramming methodology with the other available notation tools, the necessity of reaching a clear explanation of the coordination design specifications in real-time games requires the diagrams of PLATO to hold extra game-specific characteristics: illustration of the activity type, matured and precise definition of timing and attachment to the spatial attributes.

- **Illustration of the activity type:** The PAT diagram represents the game current in detailed actions for each player. For analysis purposes, visualizing a game scenario with the entire players' actions can provide us with a chart holding a considerable number of actions. Action type annotation involved in the PAT diagram provides an effective way

for mastering the scenario. In the process of coordination design, action annotation also supports designers in mapping the abstract defined interdependent actions to the final actions in the game world.

- **Precise definition of timing:** Timing as key element of coordination has been addressed clearly in PLATO. The available timing definitions in current frameworks do not match the requirements of multiplayer games and often are limited to an abstract definition of simultaneity and order. According to Table 4-1, PAT diagrams provide notation for nine different types of dependent timing, providing a flexible and solid approach for supporting the game requirements. The symbols represented in this table, play a key role in analysis and design of coordination by visualizing information about the way players interact to each other.
- **Attachment to spatial attributes:** In contrast with productivity applications, the location that the coordination activities take place is essential in games. In the design process, the LPT diagram in PLATO connects the designed coordination to the spatial attributes and in the analysis process LPT illustrates these attributes in relation with players and time.

7.3 Adopting PLATO visualization for large-scale game scenarios

Analyzing and designing coordination in large-scale game scenarios is difficult. A large scenario involves many coordination elements (such as players and objects), and there are many shared tasks, each of which can be highly collaborative, requiring complex player synchronization and activity visualization. These scenarios may consist of a numerous sequences of connected states and each stated may include several alternative milestones of coordination episodes.

PLATO visualization tools can be used to represent these complex, large-scale scenarios by dividing the scenario into connected coordination episodes. One approach to visualizing these connected coordination episodes is the concept of highgraph introduced earlier in Section 2.8. Figure 7-2 illustrates using highgraph for the representation of a (fictitious) complex collaborative scenario. The scenario considers five players coordinating in different timing contexts. While PC and PD coordinate with PA and PB from 0 to T3 in a parallel activity, PE also is coordinated with them in another parallel task but in different timing context. At T3 to T9, PC, PD, and PE wait for PA and PB to finish their activities. At T9 and by the termination of activity A, PA and PE start activity C by performing a series of soft-simultaneous actions.

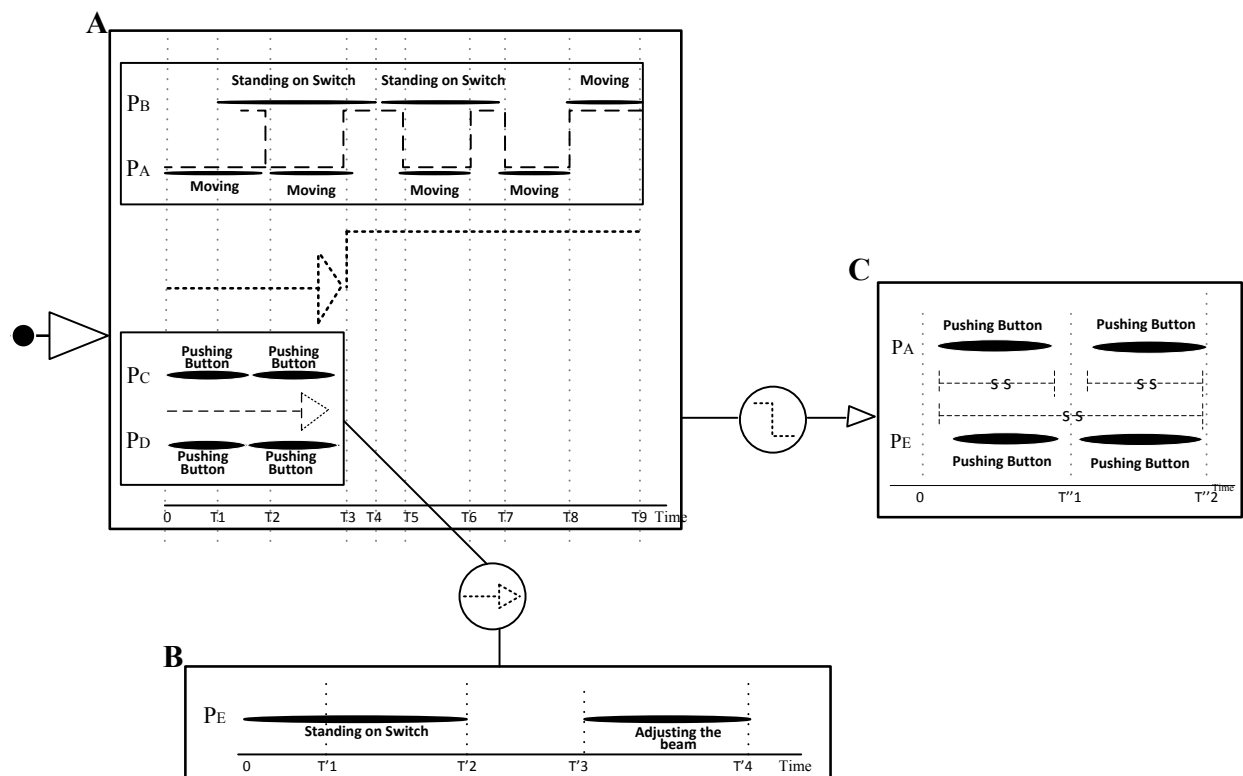


Figure 7-2. An arbitrary generalization example of PLATO notation using highgraph.

7.3.1 Game worlds with numerous players

In some genres such as massive multi-player online games, the game world consists of a large number of players participating in shared activities. This is a very different source of potential complexity than that in the example scenario shown in Figure 7-2 that was highly coordinated but only consisted of five players. However, the increase in the number of players does not necessarily require more complex coordination design. As discussed earlier, coordination plays a key role for balancing multiplayer games and overuse of player coordination results in an unbalanced and complex game experience. This implies the process of coordination design is more influenced by the game play than the scale of the game and there is always a level of constraints for coordination complexities in every game genre.

In the process of coordination design, I demonstrated the strength of the PAT diagram in the early first stages of coordination design by providing a high level of abstraction which provides a clearly illustrate coordination. Using LPT diagrams I showed the process of mapping the coordination from the PAT diagram to the spatial attributes of the game world. Finally, I demonstrated how PLATO diagram notation can be used for coordination illustration of very complex scenarios using the earlier graph-based visualization methodologies.

CHAPTER 8

CONCLUSION

8.1 Summary

In this thesis I addressed the key role of interaction designers in the process of game design and the importance of player coordination design in providing players with shared experience via collective actions. Player coordination as a key element in many multi-player real-time digital games is an important part of the design of these games and can affect the game play in several ways by changing the game balance and the level of difficulty, affecting the quality of play, and adding to the sociability of the game.

Despite the importance of player coordination design in multiplayer games, in many of the work domains studied in previous research, management of interdependencies is required to achieve productivity in contrast with multiplayer games in which the process of play and its emotional impact are more important than outcome. The current coordination frameworks are either not well matched to the realities of games or are designed for slower collaborative process.

To address the limitations and provide improved control over player coordination design, I developed a new framework (called PLATO) that can help game designers to understand, describe, design and manipulate coordination episodes. It provides a vocabulary, methodology and diagram notation for describing, analyzing, and designing coordination. I proposed two types of charts, Player-Action-Time (PAT) and Location-Player-Time (LPT) charts which are useful for elucidating the role of parallelism and simultaneity in collaborative scenarios as well as understanding how space is used in collaborative scenarios.

I demonstrated the framework's utility by describing coordination situations from existing games, and by showing how the framework of PLATO can contribute to different points within the iterative process of game design, either in earlier design phase before playtesting or in last iterations after betatesting. PLATO investigates the situation where players must coordinate, either to achieve a goal or to improve an aspect of their performance and names them as coordination episodes. It decomposes these collaborative scenarios to atomic actions describing them upon the underlying coordination elements, and refers to these inter-activity elements as interdependency factors which connect and weave several atomic actions in a certain episode.

I evaluated the strength of PLATO in designing player coordination in a real game scenario and the way it contributes to earlier design phases, and in the case study demonstrated how PLATO can contribute to the last iterations of game design. By comparing the design process with the coordination analysis methodologies, I showed the recursive procedure of coordination design in PLATO which starts with the more general definition of the scenario and leads to more specific definitions and details in each step. I demonstrated the way PLATO diagrams and tools can provide an opportunity to design player coordination and define dependencies at the very early stages without including the exact specification of the current game elements and specifications of player's actions.

8.2 Contributions

PLATO extends the ideas introduced by earlier theories of coordination to the domain of real-time multi-player games. It provides a new set of concepts that can be used to describe and characterize multi-player interactions; and it provide tools that can help designers produce better multi-player games through better control over coordination.

8.3 Future Work

We see several directions for further work with PLATO.

- First, I will test the utility and usability of the framework in studies with real game designers, to see whether the concepts and notation are valuable and comprehensible, and whether the framework can lead to definite improvements in multiplayer gameplay.
- Second, I will test the coverage of the framework and the design techniques by characterizing multiplayer interactions from several different domains, including different kinds of coupling in coordination, but also coordination episodes in different game genres (e.g., specific plays in team sports). I am also interested to see whether the ideas in the framework (e.g., the diagram notation) can be used to characterize workspace coordination as well.
- Third, I will determine whether the framework can scale up to larger coordinated episodes with more players, such as raids in World of Warcraft [68]; one main interest here is whether the diagrams become unwieldy with more people involved in the activity.
- Fourth, because the framework specifies only the coordination requirements of an episode, and not the ways that players might satisfy those requirements, I will add guidelines about awareness information that will allow other types of evaluations (e.g., whether there is adequate awareness information in the interface to carry out a coordinated task).

Coordination is a critical element of many multi-player digital games, but it is currently difficult to describe and analyze coordination episodes in designs. This work extends existing research characterizing the role of coordination in workspaces [40] to capture the more nuanced

role that coordination and timing plays in games. The framework provides a set of concepts, an analysis process, and a diagram notation, and can be used by designers to understand, evaluate, and manipulate coordination episodes both in early designs and during playtesting. PLATO provides the first comprehensive support for improving the design of a critical element of a wide variety of multiplayer games.

REFERENCES

1. Alexander, C. The origins of pattern theory: The future of the theory, and the generation of a living world. *Software, IEEE*, 16(5):71-82, 1999.
2. Benford, S., Crabtree, A., Flinham, M., Drozd, A., Anastasi, R., Paxton, M., Row-Farr, J. Can you see me now?. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 13(1):100-133, 2006.
3. Benford, S., Flinham, M., Drozd, A., Anastasi, R., Rowland, D., Tandavanitj, N., Sutton, J. Uncle Roy All Around You: Implicating the city in a location-based performance. In *Proceedings of Advances in Computer Entertainment (ACE 2004)*, 2004.
4. Björk, S., & Holopainen, J. *Patterns in game design*. Cengage Learning, 2005.
5. Björk, S., Falk, J., Hansson, R., & Ljungstrand, P. Pirates! Using the physical world as a game board. In *Proceedings of interact*, 423-430, 2001.
6. Brown, B., Bell, M. CSCW at play: 'there' as a collaborative virtual environment. In *Proceedings of the 2004 ACM conference on Computer supported cooperative work*, 350-359, November 2004.
7. Chalmers, M., Bell, M., Brown, B., Hall, M., Sherwood, S., & Tennent, P. Gaming on the edge: using seams in ubicomp games. In *Proceedings of the 2005 ACM SIGCHI International Conference on Advances in computer entertainment technology*, 306-309, 2005.
8. Chanel, G., Matias Kivikangas, J., & Ravaja, N. Physiological compliance for social gaming analysis: Cooperative versus competitive play. *Interacting with Computers*, 306-316, 2012.
9. Chen, M. G. Communication, coordination, and camaraderie in World of Warcraft. *Games and Culture*, 4(1):47-73, 2009.
10. Chen, V. H. H., Duh, H. B. L. Understanding social interaction in world of warcraft. In *Proceedings of the international conference on Advances in computer entertainment technology*, 21-24, June 2007.
11. Coleman, A. A., Colman, A. M., & Thomas, R. M. Cooperation without awareness: A multiperson generalization of the minimal social situation. *Behavioral Science*, 35(2):115-121, 1990.
12. Dickey, M. D. Game design and learning: A conjectural analysis of how massively multiple online role-playing games (MMORPGs) foster intrinsic motivation. *Educational Technology Research and Development*, 55(3):253-273, 2007.

13. Dommel, H. P., & Garcia-Luna-Aceves, J. J. Group coordination support for synchronous Internet collaboration. *Internet Computing, IEEE*, 3(2):74-80, 1999.
14. Dourish, P., & Bellotti, V. Awareness and coordination in shared workspaces. In *Proceedings of the 1992 ACM conference on Computer-supported cooperative work*, 107-114, December 1992.
15. Ducheneaut, N., & Moore, R. J. Gaining more than experience points: Learning social behavior in multiplayer computer games. In *Proceedings on human factors in computing systems (CHI2004): Extended Abstracts*, April 2004.
16. Ducheneaut, N., & Moore, R. J. The social side of gaming: a study of interaction patterns in a massively multiplayer online game. In *Proceedings of the 2004 ACM conference on Computer supported cooperative work*, 360-369, November 2004.
17. Ducheneaut, N., Yee, N., Nickell, E., & Moore, R. J. Alone together?: exploring the social dynamics of massively multiplayer online games. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, 407-416, April 2006.
18. Eccles, D. W., & Tenenbaum, G. Why an expert team is more than a team of experts: a social-cognitive conceptualization of team coordination and communication in sport. *Journal of Sport & Exercise Psychology*, 26(4):542-560, 2004.
19. Eccles, D. The coordination of labour in sports teams. *International Review of Sport and Exercise Psychology*, 3(2):154-170, 2010.
20. Enemy Territory official website. Viewed 15 November 2012, <http://enemyterritory.org/>
21. Farm Ville official blog. Viewed 15 November 2012, blog.farmville.com
22. FIFA official website. Viewed 15 November 2012, www.fifa.com
23. Fiore, S. M., Salas, E., Cuevas, H. M., & Bowers, C. A. Distributed coordination space: toward a theory of distributed team process and performance. *Theoretical Issues in Ergonomics Science*, 4(3-4):340-364, 2003.
24. Gutwin, C. The effects of network delays on group work in real-time groupware. In *ECSCW 2001*, 299-318. Springer Netherlands, January 2002.
25. Gutwin, C., & Greenberg, S. A descriptive framework of workspace awareness for real-time groupware. *Computer Supported Cooperative Work (CSCW)*, 11(3-4):411-446, 2002.
26. Gutwin, C., Penner, R., & Schneider, K. Group awareness in distributed software development. In *Proceedings of the 2004 ACM conference on Computer supported cooperative work*, 72-81, November 2004.
27. Harel, D. On visual formalisms. *Communications of the ACM*, 31(5):514-530, 1998.

28. Heath, C., & Luff, P. Collaborative activity and technological design: Task coordination in London Underground control rooms. In *Proceedings of the second conference on European Conference on Computer-Supported Cooperative Work*, 65-80. Kluwer Academic Publishers, September 1991.
29. Heath, C., Jirotko, M., Luff, P., & Hindmarsh, J. Unpacking collaboration: the interactional organisation of trading in a city dealing room. *Computer Supported Cooperative Work (CSCW)*, 3(2):147-165, 1994.
30. Huffaker, D., Wang, J., Treem, J., Ahmad, M. A., Fullerton, L., Williams, D., Contractor, N. The social behaviors of experts in massive multiplayer online role-playing games. In *Computational Science and Engineering, 2009. CSE'09. International Conference on* 4, 326-331. IEEE, 2009.
31. Huizinga, J., Homo L., International Library of Sociology G: The Sociology of Culture, 3, Routledge, 2003.
32. Hunicke, R., LeBlanc, M., & Zubek, R. MDA: A formal approach to game design and game research. In *Proceedings of the AAAI Workshop on Challenges in Game AI*, 04-04, July 2004.
33. Hutchins, E. *The technology of team navigation*, 191-220. L. Erlbaum Associates Inc, 1990.
34. Inderbitzin, M., Wierenga, S., Våljamäe, A., Bernardet, U., & Verschure, P. F. Social cooperation and competition in the mixed reality space eXperience Induction Machine XIM. *Virtual reality*, 13(3):153-158, 2009.
35. Jakobsson, M., & Taylor, T. L. The Sopranos meets EverQuest: social networking in massively multiplayer online games. In *Proceedings of the 2003 Digital Arts and Culture (DAC) conference, Melbourne, Australia*, 81-90, May 2003.
36. Kreimeier, B. The case for game design patterns, Viewed 15 November 2012. http://www.gamasutra.com/features/20020313/kreimeier_01.htm
37. Lindt, I., Ohlenburg, J., Pankoke-Babatz, U., & Ghellal, S. A report on the crossmedia game epidemic menace. *Computers in Entertainment (CIE)*, 5(1):8, 2007.
38. Mafia Wars official website. Viewed 15 November 2012, www.mafiawars.zynga.com
39. Magerkurth, C., Cheok, A. D., Mandryk, R. L., & Nilsen, T. Pervasive games: bringing computer entertainment back to the real world. *Computers in Entertainment (CIE)*, 3(3):4-4, 2005.
40. Malone, T. W., & Crowston, K. What is coordination theory and how can it help design cooperative work systems?. In *Proceedings of the 1990 ACM conference on Computer-supported cooperative work*, 357-370. ACM, September 1999.
41. Malone, T. W., & Lepper, M. R. Making learning fun: A taxonomy of intrinsic motivations for learning. *Aptitude, learning, and instruction*, 3, 223-253, 1987.

42. Manninen, T. Interaction forms and communicative actions in multiplayer games. *Game studies*, 3(1), 2003.
43. Mario official website. Viewed 15 November 2012 <http://mario.nintendo.com/>
44. Myerson, R. B. *Game theory: analysis of conflict*. Harvard University Press, 1997.
45. Nova, N., Wehrle, T., Goslin, J., Bourquin, Y., & Dillenbourg, P. Collaboration in a multi-user game: impacts of an awareness tool on mutual modeling. *Multimedia tools and Applications*, 32(2):161-183, 2007.
46. Nova, N. Awareness tools: lessons from quake-like. In *Proceedings of Playing with the Future Conference*, 167-17, 9, 2002.
47. Patton, C. M., Tatar, D., & Dimitriadis, Y. Trace Theory, Coordination Games, and Group Scribbles. In *International handbook of information technology in primary and secondary education*, 921-933. Springer US, 2008.
48. Pinelle, D., Wong, N., Stach, T., & Gutwin, C. Usability heuristics for networked multiplayer games. In *Proceedings of the ACM 2009 international conference on Supporting group work*, 169-178. ACM, May 2009.
49. Portal2 official website. Viewed 15 November 2012, <http://www.thinkwithportals.com/>
50. Rick, J., & Hsi, I. We shall be Hobbits: Learning to Succeed through Collaboration in a Game Context.
51. Rico, R., Sánchez-Manzanares, M., Gil, F., & Gibson, C. Team implicit coordination processes: A team knowledge-based approach. *Academy of Management Review*, 33(1):163-184, 2008.
52. Rocha, J. B., Mascarenhas, S., & Prada, R. Game mechanics for cooperative games. *ZON Digital Games 2008*, 72-80, 2008.
53. Rollings, A., & Adams, E. *Andrew Rollings and Ernest Adams on game design*. New Riders Pub, 2003.
54. Salen, K., & Zimmerman, E. *Rules of play: Game design fundamentals*. MIT press, 2004.
55. Schmidt, K., & Simonee, C. Coordination mechanisms: Towards a conceptual foundation of CSCW systems design. *Computer Supported Cooperative Work (CSCW)*, 5(2-3):155-200, 1996.
56. Seif El-Nasr, M., Aghabeigi, B., Milam, D., Erfani, M., Lameman, B., Maygoli, H., & Mah, S. Understanding and evaluating cooperative games. In *Proceedings of the 28th international conference on Human factors in computing systems*, 253-262. ACM, April 2010.
57. Social Structure, Britannica, Viewed 15 November 2012 <http://www.britannica.com/EBchecked/topic/551478/social-structure>.

58. Suthers, D. D., Dwyer, N., Medina, R., & Vatrappu, R. A framework for conceptualizing, representing, and analyzing distributed interaction. *International Journal of Computer-Supported Collaborative Learning*, 5(1):5-42, 2010.
59. Svensson, M. S., Heath, C., & Luff, P. Instrumental action: the timely exchange of implements during surgical operations. In *ECSCW 2007*, 41-60. Springer London, 2007.
60. Tang, T. Y., Man, C. Y., Hang, C. P., Cheuk, L. S., Kwong, C. W., Chi, Y. C., Kam, S. A study of interaction patterns and awareness design elements in a massively multiplayer online game. *International Journal of Computer Games Technology*, 2008, 5, 2008.
61. Tatar, D. G., Foster, G., & Bobrow, D. G. Design for conversation: lessons from Cognoter. *International Journal of Man-Machine Studies*, 34(2):185-209, 1991.
62. Tatar, D., Lee, J. S., & Alaloula, N. Playground games: a design strategy for supporting and understanding coordinated activity. In *Proceedings of the 7th ACM conference on Designing interactive systems*, 68-77. ACM, February 2008.
63. Tatar, D., Lin, S., & Dickey, M. Visualizing handheld-based classroom activity. In *Proceedings of the 2005 International Symposium on Collaborative Technologies and Systems*, 2005. 313-320. IEEE, May 2005.
64. Tombraider official website. Viewed 15 November 2012, <http://www.tombraider.com>
65. Vogiazou, Y., & Eisenstadt, M. Designing multiplayer games to facilitate emergent social behaviours online. *Interactive Technology and Smart Education*, 2(2):117-130, 2005.
66. Vogiazou, Y., Raijmakers, B., Geelhoed, E., Reid, J., & Eisenstadt, M. Design for emergence: experiments with a mixed reality urban playground game. *Personal and Ubiquitous Computing*, 11(1):45-58, 2006.
67. Wikipedia, *The Free Encyclopedia*. Wikimedia Foundation, Inc. 22 Jan 2012.
68. World of Warcraft official website. Viewed 15 November 2012, worldofwarcraft.us.battle.net
69. Zagal, J. P., Rick, J., & Hsi, I. Collaborative games: Lessons learned from board games. *Simulation & Gaming*, 37(1):24-40, 2006.
70. Zagal, J. P., Nussbaum, M., & Rosas, R. A model to support the design of multiplayer games. *Presence: Teleoperators & Virtual Environments*, 9(5):448-462, 2000.